.

Spatially adaptive detection of local disturbances in time series and stochastic processes on the integer lattice $\mathbb{Z}^2$

Siana Halim

Vom Fachbereich Mathematik
der Universität Kaiserslautern
zur Verleihung des akademischen Grades
Doktor der Naturwissenschaften
(Doctor rerum Naturalium, Dr. rer. nat.)
genehmigte Dissertation

1. Gutachter: Prof. Dr. Jürgen Franke
2. Gutachter: Prof. Dr. J. S. Marron

Vollzug der Promotion: 16 September 2005

D386

Abstract

In modern textile manufacturing industries, the function of human eyes to detect disturbances in the production processes which yield defective products is switched to cameras. The camera images are analyzed with various methods to detect these disturbances automatically.

There are three parts of texture analysis which are going to be studied here, i.e., image smoothing, texture synthesis and defect detection.

In the image smoothing, we shall develop a two dimensional kernel smoothing method with correlated error. Two approaches are used in synthesising texture. The first is by constructing a generalized Ising energy function in the Markov Random Field setup, and for the second, we use two dimensional bootstrap methods for semi regular texture synthesis.

We treat defect detection as multihypothesis testing problem with the null hypothesis representing the absence of defects and the other hypothesis representing various types of defects.

Acknowledgement

> But where shall I find courage? asked Frodo. That is what I chiefly need.
> Courage is found in unlikely places, said Gildor. Be of good hope! ...
>
> J.R.R. Tolkien
> The Fellowship of The Ring.

Indeed, I have found courage in many places during my study. I am grateful to Professor Franke for his support and kind help over the years, his insights that he shared during the DFG's Annual meetings always fascinated me. I was received with warmth and friendship in the Statistics and Stochastics group and I also enjoyed chatting with Frau Siegler about gardening.

To Professor J.S. Marron, I appreciate all your suggestions and comments on this thesis very much.

Many thanks to Dr. Vera Friederich for her assistance. She guided me through her work in local smoothing methods in image processing and generously gave me her program.

I found a comfortable place to work in and plenty of resources to work with in the Departmen of *Modelle und Algorithmen in der Bildverarbeitung* - Fraunhofer ITWM Kaiserslautern.

I am truly blessed through our Tuesday Bible Study group in the Kaiserslautern Church of Christ. To every member of the group, I would like to thank you all for the discussion we had and the cookies and finger food we ate together. Especially to David Emery, I thank you so much for being our teacher. Your insights forcing us to use not only our heart and our strength but also our mind to love God, helping us to focus our lives not

just in the physical space but more in the spiritual one, where the infinity lies. I thank you also for helping me in English.

I also want to thank to the *Deutsche Forschung Gemeinschaft*, for the financial support and for organizing an oppulent annual scientific meeting in the "Special Priority Program 1114", through which I was not only enriched in science but also in the German culture and culinary art.

Last but not least, I am indebted to my sisters, Dewi and Indriana, their gossips from home were always cheering me up. To my father who taught me ABC, and my mother who made me understood the meaning of zero, this work is dedicated to them.

Contents

Chapter 1

Introduction

In the modern textile industries, the function of human eyes to detect any disturbance in the processes which yield defective products are switched to cameras. The images which are shot by the camera are analyzed with various methods to detect these disturbances.

Even as Gimel'farb said that the human eyes is a natural miracle [39]

> *Our eyes solve these problems so easily that one may be puzzled why even most successful computer vision solutions rank far below in quality and processing rate. For me, human vision is a natural miracle, and more than 35 years spent in various image processing and computer vision domains suggest that it is extremely hard to bridge a gap between human and computer visual skills... But, this is still a good challenge to mimic one or another side of human vision with computational techniques, especially, because there always exist particular applied problems where computer vision can amplify or replace human vision.*
>
> *Every such problem is quite complex but just due to this reason is VERY attractive!*

However, market demands for zero defects can not afford to rely on using only human eyes as a detector, due to the problem of fatigueness. Therefore an attempt to replace human vision with computer vision is a must and, as Gimel'farb said, is very attractive !.

There are three parts of texture analysis we are going to study here, i.e., image smoothing, texture synthesis and defect detection.

Images which are taken by a camera are blended with errors that come from illumination, data transfer or data convertion. Those errors cannot be avoided and they can lead to the wrong result in defect detection. By smoothing the image, these particular errors can be reduced or even cleaned up. These methods are varied from the very traditional ones, such as, averaging filter, Gaussian filter to the sophisticated filter such as kernel smoothing, Friederich [30], adaptive weighting smoothing, Polzehl and Spokoiny [72], diffusion filtering e.g. Weickert [82].

Concerning the second problem texture synthesis. Generaly, texture is a visual property of a surface, representing the spatial information contained in object surfaces, Haindl [40]. Depending on the size of variations, textures range from purely stochastic, such as white noise, to purely regular such as chessboard. Based on the sense of its regularity structure, basically texture can be classified into three different classes random texture, semi regular texture and regular texture, as depicted in the figure. 1.1 to figure. 1.3.

The aim of texture synthesis is given a sample of texture, it should be able to generate a huge amount of data which is not exactly the same as the original, but perceived by humans to be the same texture, Julesz [52]. There exists many different methods, which basically can be grouped into two categories.The first one is model-based with main modeling tool for texture are Markov Random Fields(MRF) as in, Besag [6], Cross and Jain [18],Geman and Geman [32], Winkler [87] and spatial statistics as in Kashyap and Chellappa [53, 12]. MRF model gives a good result for a random texture, but usually does not capture the complexity of real textures. It depends on the parameters estimation,e.g., Younes [90, 91], Winkler[86], as the number of parameters increases the synthesized begin to look more realistic yet it becomes hard to estimate them. Paget in [68] developed a nonparameteric MRF to melt the hardness of these estimation procedures. Another development in the MRF direction is to apply the MRF in the multiscale, as in [67] or in the multiresolution, Lakshmanan [56, 57].

Another method is based on featuring matching, such as in Heeger and Bergen [46]. They used marginal distributions of filter outputs from a filter banks as features to match and obtained good results for stochastic textures. De Bonet's method [20] matched filter output distributions to preserved dependencies across different resolutions of the original image. Most of feature matching methods have difficulties with highly structured textures

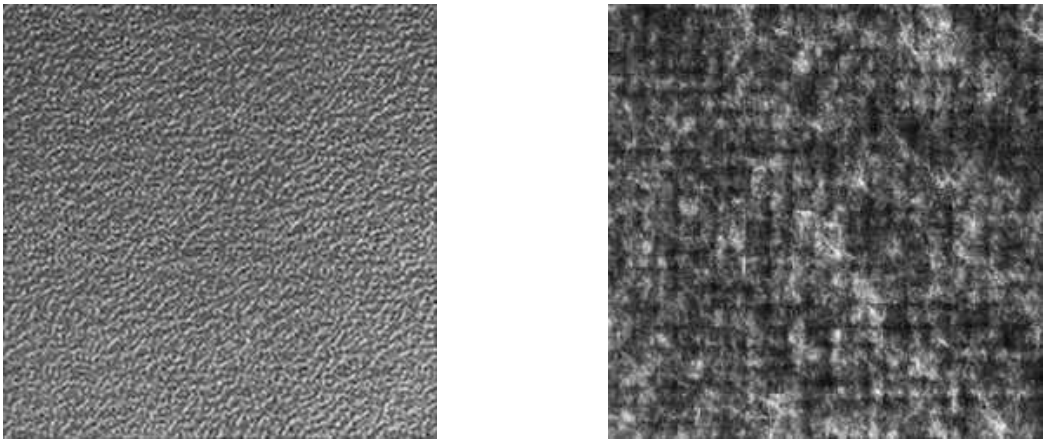


Figure 1.1: Random Texture - Simple Texture

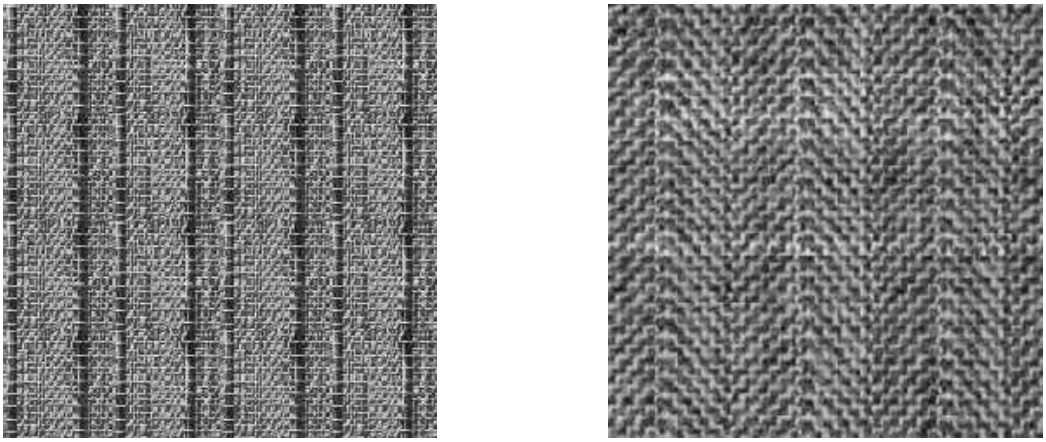


Figure 1.2: Semi Regular Texture

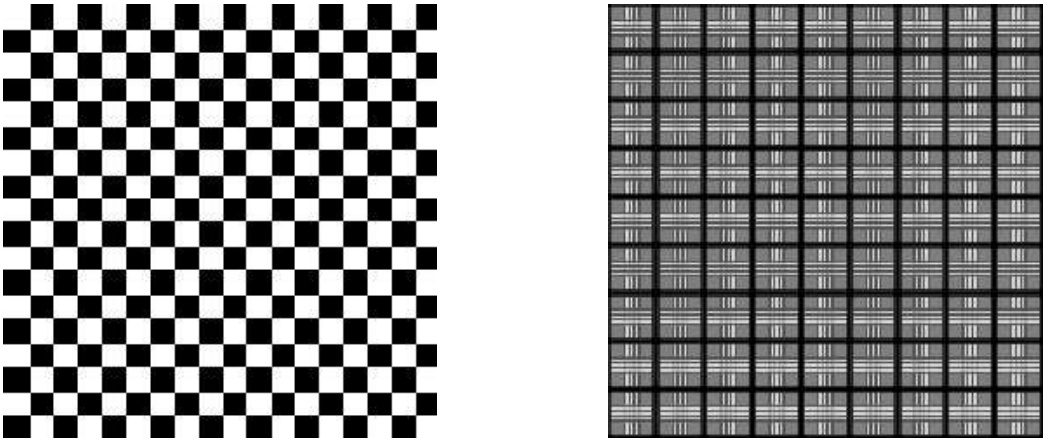


Figure 1.3: Regular Texture

and they need some tuning parameters which are done by trial and error and numbers of iteration to converge.

Zhu et.al [93] combined these two methodologies by using MRF models and also do featuring matching with their model so called FRAME (Filters, Random Field and Minimax Enthropy). They show in Wu [89] that their method is equivalent to the Julesz texture ensambles.

A recently developed method was based on a nonparametric resampling method which was started by Efros and Leung [25]. This algorithm is based on resampling from the random field directly, without constructing an explicit model for the distribution. Gimel'farb et.al [37] combine this idea with their model-based interaction map (MBIM). This map showed the inter pixel shif energy interaction, and the maximum of this energy gave the smallest sample size of the original image that represenst the pattern of the whole one.

For the last goal, i.e. in the defect detection, is to develop procedures for the detection of irregularities in gray scale images based on appropriate stochastic models for the data and on a corresponding estimation and testing theory. Most of the algorithms for the detection of irregularities in surfaces are presently based on purely heuristic arguments. Some others are using a stochastic regression model,i.e the image consists of a deterministic part wich is disturbed by residual modelled pixel-wise as independent, identically distributed random variables. The algorithms typically consists of the following steps:

1. a preprocessing of the data, i.e. smoothing the image
2. the whole picture is partitioned into small square or rectangular segments
3. for each segment, a feature or summary statistic (frequently multivariate) is calculated which measures the regularity of that part of the image
4. base on the size of regularity feature, a defect is detected or not
5. again based on the size of the regularity feature and/or on the position of adjacent defective segments the particular defect is classied.

The main differences between the algorithms originate from the use of different regularity features, where the summary statistic of choice depends on the form of the undisturbed

picture to be expected. The detection and classification is, then, based on more or less heuristic arguments, mainly comparing the components of the summary statistic with certain thresholds.

Daul et. al [19] use the local segment-wise variability of the preprocessed picture in both coordinate directions where large values hint at tha presence of defect, and the relation between both directional emperical variances help in classifying the defects. Schäl and Burkhardt [73] and Siggelkow and Schäl [79] consider local averages of certain appriopriately chosen function (e.g. monomials) of the data as regularity features which, in particular, are chosen in such a manner to make the algorithm invariant against rotation.

Other authors concentrate on measures of the local dependence of data where defetcts are assumed to break a regular dependence pattern. Chetverikov [14] considers a measure of the periodicity of the spatial autocorrelation function in polar coordinates as an appropriate regularity feature for periodic texture. This feature is etimated locally for each segment of a partition of the image, and imperfections are detected as outliers in the feature space. Alternatively, the Fourier or related transforms of the autocorrelation function is used for deriving regularity features as in Bahlmann et.al [5] or Ferryanto [26, 27].

The outline of this work will be flowed as follow.

In **Chapter 2** we present the two dimensional kernel smoothing method with correlated error and particularly will be applied to the the fabric texture which has regular characteristics. In this method we combine two major works of Friederichs in [30] and Herrmann et.al. in [47].

Chapter 3 is devout to semi regular texture synthesis. There we study the Markov Random Field(MRF), which usually are used to synthesis the random texture, parametrically as e.g.,Winkler [87],Scholz [74] or non parametrically, e.g. Paget [69], [68]. Then we propose an improved energy function in MRF to synthesis the semi regular texture.

Chapter 4 discusses the new two dimensional block bootstrap method. This idea breaking into our mind by reading the approaches of Efros,et.al in image quilting, [24], [25], and the bunch sampling approach by Gimel, et.al, [37], [92], [38]. We combine this bootstrap method with spatial error model, which has been developed by LeSage [59], and also in Luc Anselin [4], to reduce the residual error.

In **Chapter 5** we will discuss the defect detection which is treated as multihypothesis testing problem with the null hypothesis representing the absence of defects and the other hypothesis representing various types of defetcs. Departure from the models undisturbed surface, that we generated two previous chapter,i.e., texture synthesis, various summary statistics are going to be investigated which represent the regularity of the surface in a given part of the observation area. They are going to be used to test the null hypothesis of absence of defects in the particular segment againts the alternatives representing different types of defects.

Chapter 2

Regression Models with Dependent Noise for Regular Textures

The structure of regular textures is simpler than the other two, i.e, the random and semi regular ones. However, due to the illumination, conversion or digital data transfered, that noise could not be ignored. Therefore, the need of some techniques to reduce or even to clean the noise can be said crucially important in this sense.

Many techniques have been investigated for denoising images, from simple filters such as Median, Average, and Gaussian, to the more complex mathematical formulation filters, such as wavelet, nonparametric estimation procedure and diffusion filtering.

Chu, et.al [16] developed edge-preserving smoothers for image processing by defining a sigma filter, i.e., a modification of the Nadaraya Watson kernel regression estimator (see, e.g. Chu and Marron [17]). In this modification instead of using one kernel estimate, they proposed to use two types of kernel functions and two different bandwidths. The implementation of this method is for denoising images with i.i.d error, mean zero and constant variance. It does improve the edge-preserving, but is weak in terms of efficiency of noise reduction. To overcome this weakness they sugested using M smoother as an effective method of noise reduction, which needs some background in the field of robust M estimation (see, e.g., Huber [49]).

Polzehl and Spokoiny [72] developed a nonparametric estimation procedure for two-dimensional piecewise constant funtions called Adaptive Weight Smoothing(AWS). In context of image denoising, they extend the AWS procedure into a propagation-separation

approach for local likelihood [71]. The method is especially powerful for model functions having large homogeneous regions and sharp discontinuities.

In the wavelet world, Meyer and Coifman [65] constructed an adaptive basis of functions, so called Brushlets, that can be used as a tool for image compresion and directional image analysis, e.g., denoising. A year later, Candès introduced a new system for representing multivariate functions, namely, the Ridgelets [9, 10]. In [11], he used monoscale Ridgelets for representing functions that are smooth away from hyperplanes. It gives efficient representations of smooth images with smooth edges.

Friederichs [30] had investigated the denoising method based on a nonparametric regression estimation procedure, assuming the noise is independently generated and has constant variance. In this chapter, we are going to investigate that procedure with dependent noise, i.e.,instead of constant variance, the noise is characterised by autocovariance.

2.1 Regression Models

To model an image as a regression, first, we consider an equidistant grid of pixels

$$x_{ij} = \Big(\frac{i}{n} - \frac{1}{2n}, \frac{j}{n} - \frac{1}{2n}\Big) = \frac{1}{n}(i,j) - \frac{1}{2n}(1,1), \;\; i,j = 1,\ldots,n, \tag{2.1}$$

in the unit square $A = [0,1]^2$ and a function $m : [0,1]^2 \to \mathbb{R}$ to be estimated from data, i.e., the gray levels of the image as follows:

$$Y_{ij} = m(x_{ij}) + \epsilon_{ij} \quad , i,j = 1,\ldots,n, \tag{2.2}$$

where the noise is part of a stationary random field $\varepsilon_{ij}, \; -\infty < i,j < \infty$, with zero-mean and finite variance.

2.2 The Kernel Estimate of m

We use the Gasser-Müller-type kernel to estimate $m(x)$. For that purpose we decompose A into squares

$$A_{ij} = \Big\{x \in A; \; \frac{i-1}{n} \le u_1 \le \frac{i}{n}, \; \frac{j-1}{n} \le u_2 \le \frac{j}{n}\Big\}, \;\; 1 \le i,j \le n,$$

such that x_{ij} is the midpoint of A_{ij}. We consider the following local average of the observations Y_{ij} close to a given $x \in A$,

$$\hat{m}(x,h) = \sum_{i,j=1}^{n} \int_{A_{ij}} K_h(x-u)du \; Y_{ij}. \tag{2.3}$$

as an estimate of m(x). Where $K : \mathbb{R}^2 \to \mathbb{R}$ is a given kernel function, and for the bandwidth vector $h = (h_1, h_2)$, the rescaled kernel is

$$K_h(u) = \frac{1}{h_1 h_2} K\Big(\frac{u_1}{h_1}, \frac{u_2}{h_2}\Big)$$

To simplify notation, we write the index in the following way, $z = (i,j)$ such that, e.g., the model (2.2) will be in the form

$$Y_z = m(x_z) + \varepsilon_z \ , \ \ z \in I_n = \{1, \ldots, n\}^2.$$

Assumption E1: $\varepsilon_z \ , \ \ z \in \mathbb{Z}^2$, is a strictly stationary random field on the integer lattice with $\mathbb{E}\varepsilon_z = 0, \ Var \ \varepsilon_z = r(0) < \infty$ and autocovariances

$$r(z) = \mathrm{cov}(\varepsilon_{z'+z}, \varepsilon_{z'}) \ , \ \ z, z' \in \mathbb{Z}^2.$$

Assumption E2: $|r(z)| = O\Big(\frac{1}{(|z_1|+1)^\alpha (|z_2|+1)^\alpha}\Big)$ if $|z_1|, |z_2| \to \infty$ for some $\alpha > 2$.

Lemma 2.1. *Under assumptions E1, E2, there is some constant C independent of n such that*

$$\sum_{z \in I_n} \sum_{z' \notin I_n} |r(z - z')| \leq C < \infty.$$

Proof: We write $z = (i,j), \ z' = (i',j')$. If $1 \leq i \leq n$ and $i' \leq 0$ or $i' > n$, the number of pairs i, i' with $i - i' = k$ is n for any $|k| \geq n$ and $|k|$ for any $|k| \leq n$. This implies that the number of all combinations $z \in I_n, \ z' \notin I_n$ with $i - i' = k, \ j - j' = l$ is n^2 if $|k| > n$ or $|l| > n$ and $|kl|$ if $|k| \leq n$ and $|l| \leq n$. Therefore, using Assumption E2, we have for

some $c_1 > 0$

$$\begin{aligned}
& \sum_{z\in I_n}\sum_{z'\notin I_n} |r(z-z')| \\
= & \sum_{|k|>n \text{ or } |l|>n} n^2|r(k,l)| + \sum_{|k|,|l|\le n} |kl|\ |r(k,l)| \\
\le & \ c_1 \sum_{k,l=n+2}^{\infty} \frac{n^2}{(kl)^\alpha} + c_1 \sum_{k=1}^{n+1}\sum_{l=n+2}^{\infty} \frac{n^2}{(kl)^\alpha} + c_1 \sum_{k=n+2}^{\infty}\sum_{l=1}^{n+1} \frac{n^2}{(kl)^\alpha} \\
& + c_1 \sum_{k,l=1}^{n+1} \frac{(k-1)(l-1)}{(kl)^\alpha} \le C < \infty
\end{aligned}$$

as for $\alpha > 2$, e.g.,

$$\sum_{k=n+2}^{\infty} \frac{1}{k^\alpha} = O\Big(\frac{1}{n^{\alpha-1}}\Big),\ \sum_{k=1}^{\infty} \frac{1}{k^{\alpha-1}} < \infty.$$

■

Assumption M1: m is twice continuously differentiable, we use

$$m^{(\alpha,\beta)}(x) = \frac{\partial^\alpha}{\partial x_1^\beta}\frac{\partial^\alpha}{\partial x_2^\beta}\ m(x)\ ,\ \ \alpha,\beta \ge 0$$

as a notation for the derivatives of m.

Assumption K1: The kernel K is nonnegative and Lipschitz continuous with compact support $\{u;\ ||u|| \le 1\}$, and it is normed to $\int K(u)du = 1$.

Assumption K2: K is symmetric in both directions, i.e. $K(-u_1, u_2) = K(u_1, u_2) = K(u_1, -u_2)$ for all u_1, u_2.

Assumption K3: $K(u_1, u_2) = K(u_2, u_1)$ for all u_1, u_2.

In the following, let $f(\omega, \omega')$ denote the spectral density of the random field ε_{ij}, i.e. the Fourier transform of the autocovariances, which exists as a consequence of Assumption E2. In particular, we have

$$f(0,0) = \sum_{i,j=-\infty}^{\infty} r(i,j) = \sum_z r(z).$$

Proposition 2.1. *Assume (1), M1, K1-K3, E1-E2. Let* $h_1, h_2 \to 0$ *such that* $nh_1, nh_2 \to \infty$. *Then, uniformly for all* x *with* $h_1 \le x_1 \le 1-h_1, h_2 \le x_2 \le 1-h_2$, *we have*

a)If $H = \begin{pmatrix} h_1 & 0 \\ 0 & h_2 \end{pmatrix}$, *i.e., the bandwidth matrix is diagonal, then the bias is*

$$\mathbb{E}\hat{m}(x,H) - m(x) = \frac{1}{2}V_K(h_1^2 m^{(2,0)}(x) + h_2^2 m^{(0,2)}(x))^2 + o(||h||^2) + O\Big(\frac{1}{n}\Big) \tag{2.4}$$

b)If $H = \begin{pmatrix} h_{11} & h_{12} \\ h_{12} & h_{22} \end{pmatrix}$ *is arbitrary symmetric and positive definite, then the bias is*

$$\begin{aligned}\mathbb{E}\hat{m}(x,H) - m(x) = \frac{1}{2}V_K\Big((h_1^2c^2 + h_2^2s^2)m^{(2,0)}(x) + (h_1^2s^2 + h_2^2c^2)m^{(0,2)}(x) \\ + 2(h_2^2 - h_1^2)sc \;\; m^{(1,1)}\Big)^2 + o(||h||^2) + O\Big(\frac{1}{n}\Big)\end{aligned} \tag{2.5}$$

where $c = \cos\alpha, s = \sin\alpha$, *and* $h_{11} = h_1\cos^2\alpha + h_2\sin^2\alpha$, $h_{12} = (h_2 - h_1)\sin\alpha\cos\alpha$, $h_{22} = h_1\sin^2\alpha + h_2\sin^2\alpha$, $det(H) = h_1h_2$

c)and the variance will be

$$Var\,\hat{m}(x,H) = \frac{1}{n^2h_1h_2}f(0,0)Q_K + O\Big(\frac{h_1+h_2}{n^3h_1^2h_2^2}\Big) \tag{2.6}$$

where the constants $V_K = \int u_1^2K(u)du = \int u_2^2K(u)du$ *and* $Q_K = \int K^2(u)du$ *depend only on the kernel* K.

Proof:

a) and b) The bias of $\hat{m}(x,h)$ is identical to the bias for i.i.d. residuals ε_{ij}, and the result follows as a special case from Proposition 2.4 in [30].

$$\begin{aligned} \text{c)}Var\,\hat{m}(x,h) &= \sum_{z,z'\in I_n}\int_{A_z} K_h(x-u)du \int_{A_{z'}} K_h(x-v)dv \;\; \mathrm{cov}(\varepsilon_z, \varepsilon_{z'}) \\ &= \sum_{z,z'\in I_n} r(z-z')\int_{A_z}\int_{A_{z'}} K_h(x-u)\{K_h(x-v) - K_h(x-u)\}dv\;du \\ &\quad + \sum_{z,z'\in I_n} r(z-z')\frac{1}{n^2}\int_{A_z} K_h^2(x-u)du \\ &= V_1 + V_2 \end{aligned}$$

where we have used the Lebesgue measure of $A_{z'}$ is $\frac{1}{n^2}$ and

$$\frac{1}{h_1 h_2} Q_K = \int K_h^2(u)du = \int K_h^2(x-u)du$$
$$= \sum_{z \in I_n} \int_{A_z} K_h^2(x-u)du$$

where the last relation follows from the fact that $|x_1 - u_1| > h_1,\ |x_2 - u_2| > h_2$ for all $u \notin A = [0,1]^2$, therefore, $K_h(x-u) = 0$ for all $u \notin A$. Then we get

$$\begin{aligned} \Big|V_2 - \frac{f(0,0)}{n^2 h_1 h_2} Q_K\Big| &= \frac{1}{n^2}\Big|\sum_{z \in I_n}\Big\{\sum_{z' \in I_n} r(z-z') - f(0,0)\Big\}\int_{A_z} K_h^2(x-u)du\Big| \\ &\leq \frac{1}{n^4 h_1^2 h_2^2} C_K \sum_{z \in I_n}\Big|\sum_{z' \in I_n} r(z-z') - f(0,0)\Big| \end{aligned}$$

where $C_K = \max_u K(u)$. As we can write $f(0,0)$ as

$$f(0,0) = \sum_{z' \in \mathbb{Z}^2} r(z-z')$$

for arbitrary z, we get from Lemma.2.1

$$\Big|V_2 - \frac{f(0,0)}{n^2 h_1 h_2} Q_K\Big| = O\Big(\frac{1}{n^4 h_1^2 h_2^2}\Big).$$

Now, using that K is Lipschitz continuous with constant, say, L_K, we have

$$\begin{aligned} |V_1| &\leq \sum_{z, z' \in I_n} |r(z-z')| \int_{A_z}\int_{A_z'} K_h(x-u) L_K \frac{h_1+h_2}{h_1^2 h_2^2} ||u-v|| du dv \\ &\leq L_K \frac{h_1+h_2}{n^3 h_1^2 h_2^2} \sum_{z \in I_n} \int_{A_z} K_h(x-u)du \sum_{z' \in I_n} (|z_1 - z_1'| + |z_2 - z_2'| + 2)|r(z-z')| \\ &\leq L_K \frac{h_1+h_2}{n^3 h_1^2 h_2^2} \sum_{z \in I_n} \int_{A_z} K_h(x-u)du \sum_{k,l=-\infty}^{\infty} (|k| + |l| + 2) \cdot |r(k,l)| \\ &= O\Big(\frac{h_1+h_2}{n^3 h_1^2 h_2^2}\Big) \end{aligned}$$

using Assumption E2 and $\int K_h(x-u)du = 1$. For the second inequality, we have used that for $u \in A_z,\ v \in A_{z'}$ we have

$$||u-v|| \leq |u_1 - v_1| + |u_2 - v_2| \leq \frac{1}{n}(|z_1 - z_1'| + 1 + |z_2 - z_2'| + 1)$$

and the Lebesgue measure of $A_{z'}$ is $\frac{1}{n^2}$. ■

2.3 Estimating the Scaling Parameter $f(0,0)$

To estimate $f(0,0)$ we proceed similarly as in [47] and consider asymmetric differences as approximations of the residuals $\varepsilon_z,\ z \in I_n = \{1,\dots,n\}^2$. Let M be some integer with $1 \ll M \ll n$, and set

$$\mu = (M+1, M+1)\ ,\ \ \nu = (M+1, -M-1).$$

Let $|z| = |z_1| + |z_2|$ denote the ℓ_1-norm of z. Then, $|\mu| = |\nu| = 2(M+1)$, and set

$$\Delta_y = \frac{|\mu|}{|\mu|+|y|}\ ,\ \ 1-\Delta_y = \frac{|y|}{|\mu|+|y|}.$$

define

$$\begin{aligned}
\tilde{\varepsilon}_{z,y} &= Y_z - \Delta_y\, Y_{z-y} - (1-\Delta_y)\, Y_{z+\mu}\ , && y_1 \geq 0,\ y_2 > 0,\\
\tilde{\varepsilon}_{z,y} &= Y_z - \Delta_y\, Y_{z-y} - (1-\Delta_y)\, Y_{z+\nu}\ , && y_1 > 0,\ y_2 \leq 0,\\
\tilde{\varepsilon}_{z,0} &= Y_z - \frac{1}{2}(Y_{z-\mu} + Y_{z+\mu}) &&\\
\tilde{\varepsilon}_{z,y} &= Y_z - \Delta_y\, Y_{z-y} - (1-\Delta_y)\, Y_{z-\mu}\ , && y_1 \leq 0,\ y_2 < 0,\\
\tilde{\varepsilon}_{z,y} &= Y_z - \Delta_y\, Y_{z-y} - (1-\Delta_y)\, Y_{z-\nu}\ , && y_1 < 0,\ y_2 \geq 0.
\end{aligned} \tag{2.7}$$

The particular choice of the $\tilde{\varepsilon}_{z,y}$ is not important for the argument as long as it contains Y_z and Y_{z-y} and other terms which differ from $z,\ z-y$ and among themselves by at least $M+1$ in ℓ_1-norm. The above choice simplifies notation as far as possible.

For given y, let Z_y be the set of all z for which $\tilde{\varepsilon}_{z,y}$ can be calculated from the available sample $\{Y_z,\ z \in I_n\}$, e.g. $Z_y = \{z \in I_n;\ z-y,\ z+\mu \in I_n\}$ if $y_1 \geq 0,\ y_2 > 0$. Let N_y denote the number of elements in Z_y. Then, consider the following estimates of the autocovariances $r(y),\ ||y||_\infty = \max(|y_1|,|y_2|) \leq M$:

$$\begin{aligned}
\hat{r}(0) &= \frac{2}{3}\,\frac{1}{N_0}\sum_{z\in Z_0}\tilde{\varepsilon}^2_{z,0}\\
\hat{r}(y) &= -\frac{1}{2\Delta_y}\,\frac{1}{N_y}\sum_{z\in Z_y}\tilde{\varepsilon}^2_{z,y} + \Gamma_y\hat{r}(0)\ ,\ ||y||_\infty \leq M,\ y \neq 0\\
\text{with } \Gamma_y &= \frac{1}{2\Delta_y}\{1+\Delta_y^2+(1-\Delta_y)^2\}.
\end{aligned}$$

A straightforward calculation shows that these estimates are unbiased if the random field ε_z is M-dependent, i.e., $\varepsilon_z, \varepsilon_\zeta$ are independent if $|z_1 - \zeta_1| > M$ or $|z_2 - \zeta_2| > M$. Now, we approximate $f(0,0) = \sum_y r(y)$ by truncating the sum outside of $||y||_\infty \leq M$ and replacing the autocovariances by their estimates, i.e., we estimate $f(0,0)$ by

$$\hat{f}_M(0,0) = \sum_{||y||_\infty \leq M} \hat{r}(y).$$

To show consistency, we have to assume that not only the autocovariances but also the cumulants of fourth order are summable.

Assumption E3: Assume $\varepsilon_z^4 < \infty$. Let

$$\kappa(\zeta,\eta,\xi) = \mathbb{E}\ \varepsilon_z\varepsilon_{z+\zeta}\varepsilon_{z+\eta}\varepsilon_{z+\xi} - \{r(\zeta)r(\eta-\xi) + r(\eta)r(\zeta-\xi) + r(\xi)r(\zeta-\eta)\},$$

$z \in \mathbb{Z}^2$ arbitrary, $\zeta, \eta, \xi \in \mathbb{Z}^2$, denote the fourth order cumulants of the zero-mean random field $\varepsilon_z,\ z \in \mathbb{Z}^2$ and assume

$$\sum_{\zeta,\eta,\xi \in \mathbb{Z}^2} |\kappa(\zeta,\eta,\xi)| < \infty.$$

Proposition 2.2. *Let $Y_z,\ z \in I_n$, be a sample from the regression model (2.2). Assume M1, E1-E3. Then, for $M, n \to \infty$ such that $\frac{M^2}{n} \to 0$, $\hat{f}_M(0,0)$ is a consistent estimator of $f(0,0)$, and*

1. $|bias\ \hat{f}_M(0,0)| = |\mathbb{E}\hat{f}_M(0,0) - f(0,0)| = O\Big(\frac{1}{M^{\alpha-1}}\Big)$,
2. $Var\ \hat{f}_M(0,0) = O\Big(\frac{M^4}{n^2}\Big)$.

Proof:

a) A straightforward calculation shows

$$\mathbb{E}\hat{r}(0) = r(0) - \frac{4}{3}r(\mu) + \frac{1}{3}r(2\mu) \tag{2.8}$$

$$\mathbb{E}\hat{r}(y) = r(y) + \frac{1-\Delta_y}{\Delta_y}r(\mu) - \frac{4}{3}\Gamma_y r(\mu) + \frac{1}{3}\Gamma_y r(2\mu) - (1-\Delta_y)r(\mu+y) \tag{2.9}$$

$$\text{for}\ \ y_1 \geq 0,\ y_2 > 0,\ \ ||y||_\infty \leq M$$

$$\mathbb{E}\hat{r}(y) = r(y) + \frac{1-\Delta_y}{\Delta_y}r(\nu) - \frac{4}{3}\Gamma_y r(\mu) + \frac{1}{3}\Gamma_y r(2\mu) - (1-\Delta_y)r(\nu+y) \tag{2.10}$$

for $y_1 > 0,\ y_2 \le 0,\ ||y||_\infty \le M$ and analogous expressions for $y_1 \le 0$. As

$$f(0,0) = \sum_{||y||_\infty \le M} r(y) + \sum_{||y||_\infty > M} r(y)\ ,$$

we can write the bias of $\hat{f}_M(0,0)$ in the following form

$$\mathbb{E}\hat{f}_M(0,0) - f(0,0) = -\sum_{||z||_\infty > M} \beta_z r(z)$$

as all the remainder terms $r(\mu), r(\nu), r(\mu+y), r(\nu+y), r(2\mu)$ in equations (2.8)-(2.10) have indices with sup-norm $> M$. Moreover, a comparison of the coefficients shows that for some constant $c > 0$

$$|\beta_\mu|,\ |\beta_{-\mu}|, |\beta_\nu|, |\beta_{-\nu}|, |\beta_{2\mu}|, |\beta_{-2\mu}| \le cM^2$$

and $1 \le |\beta_z| \le \frac{3}{2}$ for all other $||z||_\infty > M$, where we use, e.g.,

$$\sum_{y_1 \ge 0, y_2 > 0, ||y||_\infty \le M} \frac{\Delta_y}{1-\Delta_y} = \sum_{y_1=0}^{M} \sum_{y_2=1}^{M} \frac{|y_1| + |y_2|}{2(M+1)} = O(M^2),$$

$$\sum_{y_1 \ge 0, y_2 > 0, ||y||_\infty \le M} \Gamma_y = O(M^2), \quad 0 \le \frac{\Delta_y}{1-\Delta_y},\ \Delta_y \le 1.$$

Therefore, again for some constant $c > 0$, using, e.g., $r(-\mu) = r(\mu)$,

$$\begin{aligned} |\mathbb{E}\hat{f}_M(0,0) - f(0,0)| \le & c \cdot \sum_{||y||_\infty > M} |r(z)| + cM^2(|r(\mu)| + |r(\nu)| + |r(2\mu)|) \\ = & O\Big(\frac{1}{M^{\alpha-1}}\Big) + O\Big(\frac{M^2}{M^{2\alpha}}\Big) \\ = & O\Big(\frac{1}{M^{\alpha-1}}\Big) \end{aligned}$$

as, by Assumption E2, e.g., $|r(\mu)| = O\Big(\frac{1}{(M+2)^{2\alpha}}\Big) = O\Big(\frac{1}{M^{\alpha-1}}\Big)$ and

$$\sum_{||y||_\infty > M} |r(y)| \le O\Big(\frac{1}{M^{\alpha-1}}\Big).$$

b) To get a bound on the variance of $\hat{f}_M(0,0)$, we remark that it may be written as a bilinear form in the data $Y_z,\ z \in I_n = \{1, \ldots, n\}^2$

$$\hat{f}_M(0,0) = \sum_{z,y \in I_n} s_{zy} Y_z Y_y$$

which follows immediately from the definition. By a straightforward, but lengthy calculation, using also that $N_y \sim n^2$ as $\frac{M^2}{n} \to 0$, we get

$$|s_{zy}| \begin{cases} \leq c\frac{M^2}{n^2} & \text{if } z = y, z - y = \pm\mu, z - y = \pm 2\mu \\ \leq c\frac{1}{n^2} & \text{if } ||z - y||_\infty \leq 2M + 1, z - y \neq \pm\mu \\ = 0 & \text{if } ||z - y||_\infty \geq 2(M + 1), z - y \neq \pm 2\mu \end{cases} \tag{2.11}$$

Using $Y_z = \mathbb{E}Y_z + \varepsilon_z$, we decompose

$$\begin{aligned} Var\ \hat{f}_M(0,0) &= Var\Big(\sum_{z,y\in I_n} s_{zy} Y_z Y_y\Big) \\ &= Var\Big(\sum_{z,y\in I_n} s_{zy}\{\varepsilon_z\varepsilon_y + \varepsilon_z \mathbb{E}Y_y + \varepsilon_y \mathbb{E}Y_z\}\Big) \\ &\leq 2\ Var\Big(\sum_{z,y\in I_n} s_{zy}\varepsilon_z\varepsilon_y\Big) + 2\ Var\Big(\sum_{z,y\in I_n} (s_{zy} + s_{yz})\varepsilon_y \mathbb{E}Y_z\Big) \\ &= 2V_1 + 2V_2. \end{aligned}$$

As, by Assumption M1, $\mathbb{E}Y_z = m(x_z)$ is bounded uniformly in $z \in I_n$ we have for some $c > 0$

$$\begin{aligned} |V_2| &\leq c \sum_{z,y,\zeta,\eta\in I_n} |(s_{zy} + s_{yz})(s_{\zeta\eta} + s_{\eta\zeta})|\, |r(y - \eta)| \\ &\leq c \sum_{\xi\in\mathbb{Z}^2} |r(\xi)| \sum_{z,y,\zeta\in I_n} |(s_{zy} + s_{yz})(s_{\zeta,y+\xi} + s_{y+\xi,\zeta})| \\ &= O\Big(\frac{M^4}{n^2}\Big) \end{aligned}$$

as $|r(\xi)|$ is summable by Assumption E2. For the last relation, we have used, e.g., that by (2.11)

$$\sum_{y\in I_n} s_{zy} \leq 5 \cdot c \cdot \frac{M^2}{n^2} + \sum_{||z-y||_\infty \leq 2M+1} \frac{c}{n^2} = O\Big(\frac{M^2}{n^2}\Big) \tag{2.12}$$

for all $z \in I_n$ and

$$\sum_{z,y\in I_n} s_{zy} = O\Big(\frac{M^2}{n^2}\Big) \cdot \sum_{z\in I_n} 1 = O(M^2). \tag{2.13}$$

Using a well-known identity, given, e.g., in Theorem 8.2.4 of Anderson [3],

$$\begin{aligned} V_1 &= Var(\sum_{z,y\in I_n} s_{zy}\varepsilon_z\varepsilon_y) \\ &= \sum_{z,y,\zeta,\eta\in I_n} s_{zy}s_{\zeta\eta}\{2r(z-\zeta)r(y-\eta)+\kappa(y-z,\zeta-z,\eta-z)\} \\ &= 2V_{11}+V_{12}. \end{aligned}$$

Using (2.11), we immediately have

$$\begin{aligned} V_{12} &\le c^2\frac{M^4}{n^4}\sum_{z\in I_n}\Big[\sum_{y,\zeta,\eta\in\mathbb{Z}^2}|\kappa(y-z,\zeta-z,\eta-z)|\Big] \\ &= O\Big(\frac{M^4}{n^2}\Big) \end{aligned}$$

by Assumption E3 and as I_n has n^2 elements. Also using (2.11),

$$\begin{aligned} V_{11} &\le \sum_{z,y\in I_n}|s_{zy}|\cdot\sum_{\zeta,\eta\in\mathbb{Z}^2}|s_{z+\zeta,y+\eta}|\ \ |r(\zeta)r(\eta)| \\ &\le c\frac{M^2}{n^2}\sum_{z,y\in I_n}|s_{zy}|(\sum_{\zeta\in\mathbb{Z}^2}|r(\zeta)|)^2 \\ &= O\Big(\frac{M^4}{n^2}\Big) \end{aligned}$$

as the autocovariances are summable and using (2.13). ∎

Lemma 2.2. *Assume E1-E3. Let $\alpha_{zy},\beta_z\in\mathbb{R},\ z,y\in\{1,\dots,n\}^2=I_n$, with $\alpha_{zy}=0$ if $||z-y||_\infty>L$ for some L, and the number of $z\in I_n$ with $\beta_z\neq 0$ is at most ℓ. Then,*

a) $Var\Big(\sum_{z,y\in I_n}\alpha_{zy}\varepsilon_z\varepsilon_y\Big)\le C\ n^2L^2\sup_{z,y\in I_n}\alpha_{zy}^2$

b) $Var\Big(\sum_{z\in I_n}\alpha_z\varepsilon_z\Big)\le C\ \ell\ \sup_{z\in I_n}\beta_z^2$

for some constant $C>0$ independent of n and $\alpha_{zy},\beta_z,\ \ z,y\in I_n$.

Proof:

a) Using again the cumulant identity of, e.g., Theorem 8.2.4 of Anderson [3],

$$\begin{aligned} & Var\Big(\sum_{z,y\in I_n}\alpha_{zy}\varepsilon_z\varepsilon_y\Big) \\ =& \sum_{z,y,\zeta,\eta\in I_n}\alpha_{zy}\alpha_{\zeta\eta}\{2r(z-\zeta)r(y-\eta)+\kappa(y-z,\zeta-z,\eta-z)\} \\ =& 2V_1+V_2. \end{aligned}$$

Let A denote the supremum of $|\alpha_{zy}|, z, y \in I_n$. Then, we have

$$\begin{aligned} V_2 &\le A^2\sum_{z,y,\zeta,\eta\in I_n}1_{[0,L]}(||z-y||_\infty)|\kappa(y-z,\zeta-z,\eta-z)| \\ &\le A^2\sum_{z,\zeta,\eta}\Big\{\sum_{y\in I_n}1_{[0,L]}(||y||_\infty)\cdot|\kappa(y,\zeta,\eta)|\Big\} \\ &\le A^2n^2\sum_{\zeta,\eta}\Big\{\sum_{y}1_{[0,L]}(||y||_\infty)\cdot\sum_{y}|\kappa(y,\zeta,\eta)|\Big\} \\ &\le CA^2n^2L^2 \end{aligned}$$

as $|\kappa(y,\zeta,\eta)|$ is summable by Assumption E3.

b)

$$\begin{aligned} Var\Big(\sum_{z\in I_n}\beta_z\varepsilon_z\Big) &= \sum_{z,y\in I_n}\beta_z\beta_y r(z-y) \\ &= \sum_{z,\zeta}\beta_z\beta_{z+\zeta}r(\zeta) \\ &\le \sup_{z\in I_n}|\beta_z|\sum_z|\beta_z|\sum_\zeta|r(\zeta)| \\ &\le \ell\ \sup_{z\in I_n}\beta_z^2\sum_\zeta|r(\zeta)|, \end{aligned}$$

as $|r(\zeta)|$ is summable by Assumption E2. ■

Assumption E4: Assume $\mathbb{E}|\varepsilon_z|^k < \infty$ for all $k \ge 1$ and that the k-th order cumulants of the random field $\varepsilon_z, z \in \mathbb{Z}^2$, are absolutely summable for all $k \ge 1$.

Lemma 2.3. *Assume E1, E2 and E4. Let $\alpha_{zy}, \beta_z, z, y \in I_n$ be as in Lemma 2.2. Then, for all $k \geq 1$,*

a) $\mathbb{E}(\sum_{z,y\in I_n} \alpha_{zy}\{\varepsilon_z\varepsilon_y - r(z-y)\})^{2k} \leq C_k n^{2k} L^{2k} \sup_{z,y\in I_n} |\alpha_{zs}|^{2k}$

b) $\mathbb{E}(\sum_{z\in I_n} \beta_z\varepsilon_z)^{2k} \leq C_k\ \ell\ \sup_{z\in I_n} |\beta_z|^{2k}$

for some constant C_k independent of n and $\alpha_{zy}, \beta_z,\ z+y \in I_n$.

This lemma can be shown analogously to Lemma 2.2. It has been formulated as Lemma 1 of [47] for the one-dimensional case. A direct consequence of Proposition 2.2 is:

Corollary 2.1. *Under the assumptions of Proposition.2.2, for $M = c\cdot n^{1/(1+\alpha)}$ with some constant c we get*

$$mse\ \hat{f}_M(0,0) = O(n^{-2\frac{\alpha-1}{\alpha+1}}) = o(n^{-\frac{2}{3}}).$$

Proof:
By Proposition 2.2,

$$\text{mse}\ \hat{f}_M(0,0) = O\Big(\frac{1}{M^{2(\alpha-1)}}\Big) + O\Big(\frac{M^4}{n^2}\Big).$$

For $M = c\cdot n^{-(1+\alpha)}$, the squared bias and the variance bounds are of the same order $n^{-2(\alpha-1)/(\alpha+1)}$ which is of smaller order than $n^{-2/3}$ for $\alpha > 2$. ∎

2.4 Selecting the smoothing parameter

The performance of the estimate depends crucially on the bandwidth $h = (h_1, h_2)$ of $\hat{m}(x,h)$. We consider the problem of selecting an optimal global bandwidth h_{opt} such that mean integrated squared error

$$\text{mise}\ \hat{m}(\cdot,h) = \int \mathbb{E}(\hat{m}(x,h) - m(x))^2 w_h(x)dx$$

is asymptotically minimized. w_h is a weight function with support $\{x;\ h_1 \leq x_1 \leq 1-h_1, h_2 \leq x_2 \leq 1-h_2\}$ and $\int w_h(x)dx = 1$ which we employ, for sake of simplicity, to avoid boundary effects.

Proposition 2.3. *Under the assumptions of Proposition 2.1, we have for $h_1, h_2 \to 0$ such that $nh_1, nh_2 \to \infty$*

$$mise\ \hat{m}(\cdot, h) = amise\ \hat{m}(\cdot, h) + o(h_1^4 + h_2^4) + O\Big(\frac{1}{n^2}\Big) + O\Big(\frac{h_1 + h_2}{n^3 h_1^2 h_2^2}\Big)$$

where the asymptotic mean integrated squared error is given by

$$amise\ \hat{m}(\cdot, h) = \frac{1}{4} V_K^2 \{h_1^4 I_{20} + h_2^4 I_{02} + 2h_1^2 h_2^2 I_{11}\} + \frac{f(0,0) Q_K}{n^2 h_1 h_2}$$

with $I_{kl} = \int (m^{(2,0)}(x))^k (m^{(0,2)}(x))^l w_h(x) dx,\ \ 0 \le k, l \le 2$.

This result follows from Proposition 2.1 in exactly the same manner as Proposition 2.7 of [30]. As in Proposition 2.9 of [30], an immediate consequence is

Corollary 2.2. *Under the assumptions of Proposition 2.1,*

- *amse $\hat{m}(\cdot, h)$ is minimized for $h_{ASY} = (h_{1_{ASY}}, h_{2_{ASY}})$ given by*

$$h_{1_{ASY}}^{12} = \Big(\frac{f(0,0) Q_k}{2n^2 V_k^2}\Big)^2 \frac{|m^{(0,2)}(x)|}{|m^{(2,0)}(x)|^5}$$
$$h_{2_{ASY}}^{12} = \Big(\frac{f(0,0) Q_k}{2n^2 V_k^2}\Big)^2 \frac{|m^{(2,0)}(x)|}{|m^{(0,2)}(x)|^5}$$

- *amise $\hat{m}(\cdot, h)$ is minimized for $h_{IASY} = (h_{1_{IASY}}, h_{2_{IASY}})$ given by*

$$h_{1_{IASY}}^6 = \frac{f(0,0) Q_K}{n^2 V_K^2} \left(\frac{I_{02}}{I_{20}}\right)^{\frac{3}{4}} \frac{1}{\sqrt{I_{20} I_{02}} + I_{11}}$$
$$h_{2_{IASY}}^6 = \frac{f(0,0) Q_K}{n^2 V_K^2} \left(\frac{I_{20}}{I_{02}}\right)^{\frac{3}{4}} \frac{1}{\sqrt{I_{20} I_{02}} + I_{11}} = \left(\frac{I_{20}}{I_{02}}\right)^{\frac{3}{2}} h_{a1}.$$

As the only difference between the case of i.i.d. residuals ε_z and a stationary random field ε_z is the replacement of $Var(\varepsilon_z)$ by $f(0,0)$ in the formulas for amse $\hat{m}(\cdot, h)$, amise $\hat{m}(\cdot, h)$, h_{ASY} and h_{IASY}, it follows exactly as in Remark 2.11 of [30] that the optimal bandwidth h_{opt} minimizing amse and mise $\hat{m}(\cdot, h)$ and the bandwidth h_{ASY} and h_{IASY} minimizing its asymptotic approximation are asymptotically equivalent, i.e. we have

Proposition 2.4. *Under the assumptions of Proposition 2.1*

$$h_{MSE} = h_{ASY} + o(n^{-\frac{1}{3}}).$$

$$h_{MISE} = h_{IASY} + o(n^{-\frac{1}{3}}).$$

If, additionally, m is four times continuously differentiable

$$h_{MSE} = h_{ASY} + o(n^{-\frac{1}{2}}).$$

$$h_{MISE} = h_{IASY} + o(n^{-\frac{1}{2}}).$$

Assumption K4: The kernel K is twice continuously differentiable, and all partial derivatives of order 2 are Lipschitz continuous.

2.5 Estimating the second derivative of m

The asymptotically optimal bandwidth h_a cannot be evaluated as $f(0,0)$ and the integrals I_{20}, I_{02}, I_{11} are not known. We estimate $f(0,0)$ by $\hat{f}_M(0,0)$. To get estimates of the integrals, we consider kernel estimates of the derivatives $m^{(2,0)}$ and $m^{(0,2)}$:

$$\hat{m}^{(2,0)}(x,h) = \sum_{i,j=1}^{n} \int_{A_{ij}} K_h^{(2,0)}(x-u)du \; Y_{ij}$$

$$\hat{m}^{(0,2)}(x,h) = \sum_{i,j=1}^{n} \int_{A_{ij}} K_h^{(0,2)}(x-u)du \; Y_{ij}$$

where $K_h^{(2,0)}, K_h^{(0,2)}$ denote the second derivatives of $K_h(u)$ with respect to u_1 resp. u_2, i.e.

$$K_h^{(2,0)}(u) = \frac{1}{h_1^3 h_2} \frac{\partial^2}{\partial u_1^2} K\Big(\frac{u_1}{h_1}, \frac{u_2}{h_2}\Big), \quad K_h^{(0,2)}(u) = \frac{1}{h_1 h_2^3} \frac{\partial^2}{\partial u_2^2} K\Big(\frac{u_1}{h_1}, \frac{u_2}{h_2}\Big).$$

Plugging the derivative estimates into the integrands, we get

$$\hat{I}_{kl}(h) = \int (\hat{m}^{(2,0)}(x,h))^k (\hat{m}^{(0,2)}(x,h))^l w_h(x)dx \;, \; 0 \le k,l \le 2.$$

In practice, we would approximate the integral by an average over the grid points x_{ij} which makes an asymptotically negligible difference only. Now, we are confronted with the problem of choosing the bandwidth $h = (h_1, h_2)$ for the derivative estimates $\hat{m}^{(2,0)}, \hat{m}^{(0,2)}$, which, for sake of simplicity, we have chosen as equal. We proceed iteratively using the plug-in algorithm of [31] in the two-dimensional form of [30].

2.6 Data-Adaptive Smoothing Parameters

2.6.1 Two Bandwidth Parameters

(I0) Initial step: Estimate $f(0,0)$ by $\hat{f}_M(0,0)$ and set $\hat{h}_1^{(0)} = n^{-1}, \hat{h}_2^{(0)} = n^{-1}$.

(I1) Iteration from $i-1$ to i, $i = 1, \ldots, i^*$: Inflate the previous bandwidth $\hat{h}^{(i-1)}$ by a factor n^β:

$$\hat{h}_\beta^{(i-1)} = n^\beta \hat{h}^{(i-1)} \quad \text{and set}$$

$$\hat{h}_1^{(i)} = \left(\frac{\hat{f}_M(0,0) Q_K}{n^2 V_K^2} \right)^{\frac{1}{6}} \left(\frac{\hat{I}_{02}(\hat{h}_\beta^{(i-1)})}{\hat{I}_{20}(\hat{h}_\beta^{(i-1)})} \right)^{\frac{1}{8}} \left(\frac{1}{\sqrt{\hat{I}_{20}(\hat{h}_\beta^{(i-1)}) \hat{I}_{02}(\hat{h}_\beta^{(i-1)})} + \hat{I}_{11}(\hat{h}_\beta^{(i-1)})} \right)^{\frac{1}{6}}$$

$$\hat{h}_2^{(i)} = \left(\frac{\hat{I}_{20}(\hat{h}_\beta^{(i-1)})}{\hat{I}_{02}(\hat{h}_\beta^{(i-1)})} \right)^{\frac{1}{4}} \hat{h}_1^{(i)}$$

(I2) Iteration from $j = 1, ..., j^*$ repeat

$$\hat{h}_1^{i^*+j}(x) = \left(\frac{\hat{f}_M(0,0) Q_k}{n^2 V_k^2} \frac{1}{S_1} \right)^{\frac{1}{6}}$$

$$\hat{h}_2^{i^*+j}(x) = \left(\frac{\hat{f}_M(0,0) Q_k}{n^2 V_k^2} \frac{1}{S_2} \right)^{\frac{1}{6}}$$

where S_1 is the maximum of $2 \frac{|\hat{m}^{(2,0)}(x; n^\beta \hat{H}^{(i^*+j-1)})|^{\frac{5}{2}}}{|\hat{m}^{(0,2)}(x; n^\beta \hat{H}^{(i^*+j-1)})|^{\frac{1}{2}}}$ itself or its smoothed version (with bandwidths of the previous iteration step) and S_2 is the maximum of $2 \frac{|\hat{m}^{(0,2)}(x; n^\beta \hat{H}^{(i^*+j-1)})|^{\frac{5}{2}}}{|\hat{m}^{(2,0)}(x; n^\beta \hat{H}^{(i^*+j-1)})|^{\frac{1}{2}}}$ itself or its smoothed version, where

$$\hat{H}^{(i-1)} = \begin{pmatrix} c_1 \hat{h}_1^{(i-1)} & 0 \\ 0 & c_2 \hat{h}_2^{(i-1)} \end{pmatrix}$$

c_1 and c_2 are constants. As in the [30], in step 2 and 3 bandwidths are restricted to the interval $[\frac{1}{2\sqrt{n}}, \frac{1}{2}]$. If they happend to be outside, then their values are set to be $\frac{1}{2\sqrt{n}}$,or $\frac{1}{2}$ resp.

2.6.2 Three Bandwidth Parameters

For three bandwidth parameters the algorithm does not change for the first and second step. In step 3 estimators for α, h_1, h_2 are obtained using the AMSE-optimal formulas, (2.5). However, the main problem in these estimators is the possibility of undefiniteness in the Hessian matrix of the AMSE, i.e.

$$\begin{aligned} AMSE(x,H) &= V_k^2\Big((h_1^2c^2+h_2^2s^2)m^{(2,0)}+(h_1^2s^2+h_2^2c^2)m^{(0,2)} \\ &\qquad +2(h_2^2-h_1^2)m^{(1,1)})sc\Big)^2+\tfrac{f(0,0)Q_k}{2n^2V_k^2} \end{aligned} \tag{2.14}$$

$$\alpha_{ASY}=\begin{cases}0, & \text{if } m^{(0,2)}(x)-m^{(2,0)}(x)=0\\ \frac{1}{2}\arctan\frac{2m^{(1,1)}(x)}{m^{(0,2)}(x)-m^{(2,0)}(x)}, & \text{otherwise}\end{cases} \tag{2.15}$$

$$h_{1ASY}=\Big(\frac{f(0,0)Q_k}{2n^2V_k^2}\Big)^{\frac{1}{6}}\frac{|z|^{\frac{1}{12}}}{|u|^{\frac{5}{12}}}$$

$$h_{2ASY}=\Big(\frac{f(0,0)Q_k}{2n^2V_k^2}\Big)^{\frac{1}{6}}\frac{|u|^{\frac{1}{12}}}{|z|^{\frac{5}{12}}}$$

with

$$u=\begin{pmatrix}\cos\alpha & \sin\alpha\end{pmatrix}\begin{pmatrix}m^{(2,0)} & m^{(1,1)}\\ m^{(1,1)} & m^{(0,2)}\end{pmatrix}\begin{pmatrix}\cos\alpha\\ \sin\alpha\end{pmatrix}$$

$$z=\begin{pmatrix}\cos\alpha & -\sin\alpha\end{pmatrix}\begin{pmatrix}m^{(2,0)} & m^{(1,1)}\\ m^{(1,1)} & m^{(0,2)}\end{pmatrix}\begin{pmatrix}\cos\alpha\\ -\sin\alpha\end{pmatrix}$$

if

$$M=\begin{pmatrix}m^{(2,0)} & m^{(1,1)}\\ m^{(1,1)} & m^{(0,2)}\end{pmatrix} \tag{2.16}$$

is indefinite, then we are running into the problem mentioned, since we are going to get $sign(u)\neq sign(z)$ or $u=0$ or $z=0$, and this causes the bandwidth h_1 and h_2 are going to be out of control. There are several proposals to overcome this problem, the easiest way is using the modified Cholesky decomposition, e.g. [36], a method to transform the indefinite matrix into definite matrix. However, for a 2 x 2 matrix as our case, this

decomposition has the tendency to make the component of the matrix bigger in order to get the determinant matrix greater than zero.

Terrell and Scott in [80], in the case of a multivariate balloon estimator given by

$$\hat{f} = \frac{1}{n|B(x)|}\sum_{i=1}^{n} K(B(x)^{-1}(x - x_i))$$

where B is a scaling matrix dependent upon the point estimation, K is the kernel, showed the asymptotic bias of this estimator can be approximated by

$$|Bias(x)| = \frac{b^2}{A} tr(A^t S_x A)$$

where tr denote the trace of matrix and S_x is the Hessian matrix. Here, the scaling matrix B at some estimation point x can be rewritten as bA where $|A| = 1$. We can see that the AMSE of this estimator also has a problem with indefiniteness. In the case of S_x indefinite, they suggest to choose A, so that the AMSE at these points can be made negligible compared to those points which have definiteness. The justification lies in the direction of the curvature of f. When the curvature is upward in some directions and downwards in others, A can be chosen to make the positive and negative terms in the bias of the appropriate order cancel.

But, in our case, the AMSE of the changes of that vector means to perturb the angle, i.e., α, and the angle is formulated by the derivative of the kernel itself, see, (2.15). Therefore we propose the following way to transform the indefiniteness out of matrix M.

Proposition 2.5. *Let α follows(2.15) and M be given by (2.16) indefinite. Let $\theta = \frac{m^{(0,2)} - m^{(2,0)}}{\sqrt{4(m^{(1,1)})^2 + (m^{(0,2)} - m^{(2,0)})^2}}$*

- *if $m^{(2,0)} < 0$ and $m^{(0,2)} > 0$ then intuitively we want to perturb the value of $m^{(2,0)}$ to avoid indefiniteness, and we propose the following changes*

$$m^{(2,0)^*} = \begin{cases} m^{(0,2)} + \frac{2m^{(1,1)}\theta}{\sqrt{1-\theta^2}}, & \text{if } m^{(1,1)} > 0 \\ m^{(0,2)} - \frac{2m^{(1,1)}\theta}{\sqrt{1-\theta^2}}, & \text{if } m^{(1,1)} < 0 \end{cases}$$

- *Similarly, if $m^{(2,0)} > 0$ and $m^{(0,2)} < 0$*

$$m^{(0,2)^*} = \begin{cases} m^{(2,0)} + \frac{2m^{(1,1)}\theta}{\sqrt{1-\theta^2}}, & \text{if } m^{(1,1)} > 0 \\ m^{(2,0)} - \frac{2m^{(1,1)}\theta}{\sqrt{1-\theta^2}}, & \text{if } m^{(1,1)} < 0 \end{cases}$$

where $m^{(2,0)^*}$ and $m^{(0,2)^*}$ are the new value of $m^{(2,0)}$ and $m^{(0,2)}$ respectively. $m^{(2,0)^*}$ and $m^{(0,2)^*}$ are positive.

Proof.

$$\begin{aligned}
\alpha &= \frac{1}{2}\arctan\frac{2m^{(1,1)}(x)}{m^{(0,2)}(x) - m^{(2,0)}(x)} \\
&\Leftrightarrow \tan(2\alpha) = \frac{2m^{(1,1)}}{m^{(0,2)} - m^{(2,0)}} \\
&\Leftrightarrow \cos^2(2\alpha) = \frac{(m^{(0,2)} - m^{(2,0)})^2}{4(m^{(1,1)})^2 + (m^{(0,2)} - m^{(2,0)})^2} \\
&\Leftrightarrow \alpha = \frac{1}{2}\arccos\Big(\frac{m^{(0,2)} - m^{(2,0)}}{\sqrt{4(m^{(1,1)})^2 + (m^{(0,2)} - m^{(2,0)})^2}}\Big)
\end{aligned}$$

We let $\theta = \frac{m^{(0,2)} - m^{(2,0)}}{\sqrt{4(m^{(1,1)})^2 + (m^{(0,2)} - m^{(2,0)})^2}}$ and by direct substitution of $m^{(2,0)^*}$ to $m^{(2,0)}$ or $m^{(0,2)^*}$ to $m^{(2,0)}$ we get the result. □

The geometric interpretation of this proposition is that we rotate the smoothing ellipse by $\frac{\pi}{2}$. Since by this substitution we rotate θ to be $-\theta$, and we know that the $\arccos(-\theta) = \arccos(\theta) + \pi$ this implies $\alpha^* = \frac{1}{2}(\arccos(\theta) + \pi) = \alpha + \frac{\pi}{2}$.

Remark 2.1. *This proposition does not guarantee that we get definiteness in one step. Therefore in the next steps, we are going to tune the α small enough until $det(M) > 0$.*

Corollary 2.3. *The new bandwidth will be*

$$\begin{aligned}
h_1^* &= K\Big(\Big|\frac{u+a}{z+b}\Big|^{\frac{3}{2}}\frac{1}{2(u+a)(z+b)}\Big)^{\frac{1}{6}} \\
h_2^* &= K\Big(\Big|\frac{z+b}{u+a}\Big|^{\frac{3}{2}}\frac{1}{2(u+a)(z+b)}\Big)^{\frac{1}{6}}
\end{aligned}$$

where

$$\begin{aligned}
u &= \sin^2\alpha m^{(2,0)} + \cos^2\alpha m^{(0,2)} + 2\sin\alpha\cos\alpha \quad m^{(1,1)} \\
z &= \cos^2\alpha m^{(2,0)} + \sin^2\alpha m^{(0,2)} - 2\sin\alpha\cos\alpha \quad m^{(1,1)}
\end{aligned}$$

and

- *if* $m^{(2,0)} < 0, m^{(0,2)} > 0$

$$a = m^{(0,2)} - m^{(2,0)} - 4\sin\alpha\cos\alpha \quad m^{(1,1)}$$
$$b = m^{(0,2)} - m^{(2,0)} + 4\sin\alpha\cos\alpha \quad m^{(1,1)}$$

- *if* $m^{(2,0)} > 0, m^{(0,2)} < 0$

$$a = m^{(2,0)} - m^{(0,2)} - 4\sin\alpha\cos\alpha m^{(1,1)}$$
$$b = m^{(2,0)} - m^{(0,2)} + 4\sin\alpha\cos\alpha m^{(1,1)}$$

and the differences are

$$h_1^* - h_1 = O(z^{-5/12})$$
$$h_2^* - h_2 = O(u^{-5/12})$$

provided $(u+a) > 0$ *and* $(z+b) > 0$

Proof. Direct calculation from the definition of AMSE (2.14), and substitution of α with α^* and $m^{(2,0)}, m^{(1,1)}$ with $m^{(2,0)^*}, m^{(1,1)^*}$ respectively.

$$\begin{aligned}
h_1^* - h_1 &= K\Big[\Big(\Big|\frac{u+a}{z+b}\Big|^{3/2}\frac{1}{2(u+a)(z+b)}\Big)^{1/6} - \Big(\Big|\frac{u}{z}\Big|^{3/2}\frac{1}{uz}\Big)\Big]^{1/6} \\
&= K\Big[\frac{(u+a)^{1/12}}{2^{1/6}(z+b)^{5/12}} - \frac{u^{1/12}}{z^{5/12}}\Big] \\
&\leq K\Big[\frac{(u^{1/12}+a^{1/12})}{2^{1/6}(z^{5/12}+b^{5/12})} - \frac{u^{1/12}}{z^{5/12}}\Big] \\
&= K\Big[\frac{u^{1/12}}{2^{1/6}(z^{5/12}+b^{5/12})} - \frac{u^{1/12}}{z^{5/12}} + \frac{a^{1/12}}{2^{1/6}(z^{5/12}+b^{5/12})}\Big] \\
&\leq \frac{K_1}{z^{5/12}+b^{5/12}} \\
&= O(z^{-5/12})
\end{aligned}$$

By the same way we get $h_2^* - h_2 = O(u^{-5/12})$ □

2.7 The Asymptotic properties

We get the following main result analogously to Theorem 4.1 of [30] under two additional assumptions.

Assumption M2: $m(x)$ is four times continuously differentiable, and all partial derivatives of order 4 are Hölder continuous with some exponent $\delta > 0$, i.e. for some $c > 0$ and all x, x' and $0 \leq i, j \leq 4$ with $i + j = 4$

$$\left| \frac{\partial^4}{\partial x_1^i \partial x_2^j} m(x) - \frac{\partial^4}{\partial x_1^i \partial x_2^j} m(x') \right| \leq c||x - x'||^\delta.$$

Assumption W1: The weight function $w_h(x)$ is twice continuously differentiable in x, and for any fixed $\delta > 0$ we have uniformly in $\{x; \delta \leq x_1, x_2 \leq 1 - \delta\}$ that $w_h(x) \to 1$ and all partial derivatives of $w_h(x)$ up to order 2 converge to 0.

Theorem 2.1. *Assume that the data are generated by model in (2.3) and that the assumptions M1, M2, K1-K4, W1, E1, E2 with $\alpha \geq 3$, E4, are satisfied. Let $\beta = \frac{1}{6}$ in the iterative step (I1). Assume furthermore that $I_{20}, I_{02} \neq 0$ and $\sqrt{I_{20} I_{02}} + I_{11} \neq 0$. Then,*

$$\hat{h}^{(7)}_{MISE} = h_{MISE-opt}\big(1 + O(n^{-\frac{1}{3}}) + O_p(n^{-1})\big)$$

$$\hat{h}^{(7)}_{MSE} = h_{MSE-opt}\big(1 + O(n^{-\frac{1}{3}}) + O_p(n^{-\frac{1}{2}})\big)$$

as in the case of i.i.d. noise if we choose $M = c \cdot n^{1/(1+\alpha)}$ for some constant $c > 0$, $\alpha \geq 3$. Moreover, further iterations $\hat{h}^{(i)}, i > 7$, do not improve the estimates further.

Proof. The proof of the bandwidth estimator, follows strictly the proof of Theorem 4.1. from [30]. Instead of using the Whittle 's inequality [84], we apply

$$\mathbb{E}\Big(\sum_{k=1}^{N}\sum_{i=1}^{N} c_{ik}(X_i X_j - E(X_i X_j))\Big)^{2\alpha} \leq C_\alpha \Big(\sum_{k=1}^{N}\sum_{i=1}^{N} c_{ik}^2\Big)^{\alpha}$$

$$\mathbb{E}\Big(\sum_{i=1}^{n} \beta_i X_i\Big)^{2\alpha} \leq c_\alpha \Big(\sum_{i=1}^{n} \beta^2\Big)^{\alpha}$$

where α is a positive integer, and c_{ik} are real numbers, and $X_i, i = 1, ..., N$, independent random variables with expectation zero. The constant C_α is positive and depends on the moments of X_i. We are using Lemma 2.3 to replace the independent random variables in Whittle's result with the correlated ones.

In the proof, bandwidths h_1 dan h_2 are considered which are random variables and have values in the interval $[\frac{1}{2n}, \delta]$ such that there are no boundary effects to be considered. In some places it is assumed that nonrandom bandwidths $\tilde{h}_1$ and $\tilde{h}_2$ exist which approximate the random h_1 and h_2 in the following way

$$h_1 = \tilde{h}_1(1 + o_p(N^\gamma)) \tag{2.17}$$

$$h_2 = \tilde{h}_2(1 + o_p(N^\gamma)) \tag{2.18}$$

for some $\gamma > 0$ and $N = n^2$. Further, it is assumed that $\tilde{h}_1$ and $\tilde{h}_2$ are of the same order and converge to zero. The existence of such $\tilde{h}_1$ and $\tilde{h}_2$ and all other assumptions on h_1 and h_2 will be ensured during the iteration steps.

First consider,

$$\begin{aligned}
\Big(\hat{m}^{(2,0)}(x,h)\Big)^2 &= \Big[\sum_{z=1}^{N} \int_{A_z} K_h^{(2,0)}(x-u)du\ (m(x_z)+\epsilon_z)\Big]^2 \\
&= \Big[\sum_{z=1}^{N} \int_{A_z} K_h^{(2,0)}(x-u)du\ m(x_z)\Big]^2 && \text{(2.19a)} \\
&+ 2\sum_{z=1}^{N}\sum_{z'=1}^{N} \int_{A_z}\int_{A_{z'}} K_h^{(2,0)}(x-u)K_h^{(2,0)}(x-v)du\ dv\ m(x_z)\epsilon_{z'} && \text{(2.19b)} \\
&+ \sum_{z=1}^{N}\sum_{z'=1}^{N} \int_{A_z}\int_{A_{z'}} K_h^{(2,0)}(x-u)K_h^{(2,0)}(x-v)du\ dv\ \epsilon_z\epsilon_{z'} && \text{(2.19c)}
\end{aligned}$$

Denote (2.19a) by $\mathcal{B}^2(x,h)$, (2.19b) by $\mathcal{M}(x,h)$ and decompose it in two parts, that is,

$$\mathcal{M}_1(x,h) = 2\sum_{z=1}^{N}\sum_{z'=1}^{N} \int_{A_z}\int_{A_{z'}} K_h^{(2,0)}(x-u)K_h^{(2,0)}(x-v)\big(m(x_z)-m(x_u)\big)du\ dv\epsilon_{z'} \tag{2.20a}$$

$$\mathcal{M}_2(x,h) = 2\sum_{z=1}^{N}\sum_{z'=1}^{N} \int_{A_z}\int_{A_{z'}} K_h^{(2,0)}(x-u)K_h^{(2,0)}(x-v)m(x_u)du\ dv\epsilon_{z'} \tag{2.20b}$$

Denote (2.19c) by $\mathcal{V}(x,h)$ and decompose it as follows

$$\mathcal{V}_1(x,h) = \sum_{z=1}^{N}\sum_{z'=1}^{N}\int_{A_z}\int_{A_{z'}} K_h^{(2,0)}(x-u)K_h^{(2,0)}(x-v)du\ dv\,\mathrm{cov}(\epsilon_z,\epsilon_{z'}) \tag{2.21a}$$

$$\mathcal{V}_2(x,h) = \sum_{z=1}^{N}\sum_{z'=1}^{N}\int_{A_z}\int_{A_{z'}} K_h^{(2,0)}(x-u)K_h^{(2,0)}(x-v)du\ dv\Big(\epsilon_z\epsilon_{z'} - \mathrm{cov}(\epsilon_z,\epsilon_{z'})\Big) \tag{2.21b}$$

We proceed first to calculate $\mathcal{V}_1$ following proposition 2.1

$$\begin{aligned}
\mathcal{V}_1(x,h) &= \sum_{z=1}^{N}\sum_{Z'=1}^{N}\int_{A_z}\int_{A_{z'}} K_h^{(2,0)}(x-u)K_h^{(2,0)}(x-v)du\ dv\,\mathrm{cov}(\epsilon_z,\epsilon_{z'}) \\
&= \sum_{z,z'\in I_N} r(z-z')\int_{A_z}\int_{A_{z'}} K_h^{(2,0)}(x-u)\Big\{K_h^{(2,0)}(x-v) - K_h^{(2,0)}(x-u)\Big\}du\ dv \\
&+ \sum_{z,z'\in I_N} r(z-z')\frac{1}{N}\int_{A_z}\Big(K_h^{(2,0)}(x-u)\Big)^2 du \\
&= V_1 + V_2
\end{aligned}$$

We use

$$\begin{aligned}
\frac{1}{h_1^3 h_2}Q_k^{(2,0)} &= \int\Big(K^{(2,0)}(u)\Big)^2 du = \int\Big(K^{(2,0)}(x-u)\Big)^2 du \\
&= \sum_{z\in I_N}\int_{A_z}\Big(K^{(2,0)}(x-u)\Big)^2 du
\end{aligned}$$

$$\begin{aligned}
\Big|\mathcal{V}_2 - \frac{f(0,0)}{Nh_1^3h_2}Q_k^{(2,0)}\Big| &= \frac{1}{N}\Big|\sum_{z\in I_N}\Big\{\sum_{z'\in I_N} r(z-z') - f(0,0)\Big\}\int_{A_z}\Big(K_h^{(2,0)}(x-u)\Big)^2 du\Big| \\
&\le \frac{1}{N}\lambda(A_z)\max_u\Big(K_h^{(2,0)}(x-u)\Big)^2\sum_{z\in I_N}\Big|\sum_{z'\in I_N} r(z-z') - f(0,0)\Big| \\
&= \frac{1}{N^2h_1^6h_2^2}\max_u\Big(K_h^{(2,0)}(x-u)\Big)^2 \\
&= O\Big(\frac{1}{N^2h_1^6h_2^2}\Big)
\end{aligned}$$

since $\sum_{z\in I_N}\Big|\sum_{z'\in I_N} r(z-z') - f(0,0)\Big| < \infty$ by Lemma 2.1 and $\max_u\Big(K_h^{(2,0)}(x-u)\Big)^2 = C_k^{(2,0)}$ a constant.

$$\begin{aligned}
|V_1| &\le \sum_{z,z'\in I_N}\Big|r(z-z')\Big|\int_{A_z}\int_{A_{z'}} K_h^{(2,0)}\Big\{K_h^{(2,0)}(x-v) - K_h^{(2,0)}(x-u)\Big\}du\ dv \\
&\le L_k\frac{h_1+h_2}{N^{3/2}h_1^2h_2^2}\sum_{z\in I_N}\int_{A_z} K_h^{(2,0)}(x-u)du \sum_{k,l=-\infty}^{\infty}(|k|+|l|+2)|r(k,l)| \\
&= o(1)
\end{aligned}$$

Since for $N\to\infty, \sum_{z\in I_N}\int_{A_z} K_h^{(2,0)}(x-u)du = \int K_h^{(2,0)}(x-u)du = 0$ due the the definition of kernel derivative, and $\sum_{k,l=-\infty}^{\infty}(|k|+|l|+2)\ |r(k,l)| \le C < \infty$ due to assumption E2.

We get

$$\mathcal{V}_1 = \frac{f(0,0)}{Nh_1^3h_2}Q_k^{(2,0)} + O\Big(\frac{1}{N^2h_1^6h_2^2}\Big) + o(1)$$

if $\frac{1}{h_1} = o_p(n), \frac{1}{h_2} = o_p(n)$ and $h_1, h_2 = o_p(1)$ then

$$\mathcal{V}_1 = \frac{f(0,0)}{Nh_1^3h_2}Q_k^{(2,0)} + O(N^2) + o(1) \tag{2.22}$$

$$\begin{aligned}
\mathcal{V}_2(x,h) &= \sum_{z=1}^{N}\sum_{z'=1}^{N}\int_{A_z}\int_{A_{z'}} K_h^{(2,0)}(x-u)K_h^{(2,0)}(x-v)du\ dv\Big(\epsilon_z\epsilon_{z'} - \mathrm{cov}(\epsilon_z,\epsilon_{z'})\Big) \\
&= \sum_{z=1}^{N}\sum_{z'=1}^{N} a_{zz'}(x,h)\Big(\epsilon_z\epsilon_{z'} - r(z-z')\Big)
\end{aligned}$$

where

$$\begin{aligned}
a_{zz'}(x,h) &= \int_{A_z}\int_{A_{z'}} K_h^{(2,0)}(x-u)K_h^{(2,0)}(x-v)du\ dv \\
&= \frac{1}{h_1^6h_2^2}\int_{A_z}\int_{A_{z'}} K^{(2,0)}\left(H^{-1}\begin{pmatrix}x_1-u_1\\x_2-u_2\end{pmatrix}\right)K^{(2,0)}\left(H^{-1}\begin{pmatrix}x_1-v_1\\x_2-v_2\end{pmatrix}\right)du\ dv
\end{aligned}$$

We use the discrete lattice approximation of the bandwidths h_1, h_2, that is, $H_{j,N}^{\rho_j},\ j=1,2$ is a set of bandwidth with the following properties $\#H_{j,n}^{\rho_j} = N^{\rho_j}$, $\max_{h_j\in[\frac{1}{2n},\frac{1}{2}]}\min_{\bar{h}_j\in H_{j,N}^{\rho_j}}|h_j - \bar{h}_j| \le N^{\rho_j}$, to calculate

$$\Big|N\bar{h}_1^5\bar{h}_2\sum_{z=1}^{N}\sum_{z'=1}^{N}a_{zz'}(x,\bar{h})\Big(\epsilon_z\epsilon_{z'}-\mathrm{cov}(\epsilon_z,\epsilon_{z'})\Big)\Big|$$

$$\leq \sup_{\bar{h}_j\in H_{j,N}^{\rho_j}}\Big|N\bar{h}_1^5\bar{h}_2\sum_{z=1}^{N}\sum_{z'=1}^{N}a_{zz'}(x,\bar{h})\Big(\epsilon_z\epsilon_{z'}-\mathrm{cov}(\epsilon_z,\epsilon_{z'})\Big)\Big| \tag{2.23a}$$

$$+\inf_{\bar{h}_j\in H_{j,N}^{\rho_j}}\Big|Nh_1^5h_2\sum_{z=1}^{N}\sum_{z'=1}^{N}a_{zz'}(x,h)\Big(\epsilon_z\epsilon_{z'}-\mathrm{cov}(\epsilon_z,\epsilon_{z'})\Big)$$

$$-N\bar{h}_1^5\bar{h}_2\sum_{z=1}^{N}\sum_{z'=1}^{N}a_{zz'}(x,\bar{h})\Big(\epsilon_z\epsilon_{z'}-\mathrm{cov}(\epsilon_z,\epsilon_{z'})\Big)\Big| \tag{2.23b}$$

Equation (2.23b) can be solved exactly as equation(46) in [30] and we get $o_p(1)$.

For all positive real numbers ϵ and η and for all positive integers α

$$\Pr\Big(\sup_{\bar{h}_j\in H_{j,N}^{\rho_j}}\Big|N\bar{h}_1^5\bar{h}_2\sum_{z=1}^{N}\sum_{z'=1}^{N}a_{zz'}(x,\bar{h})\Big(\epsilon_{zz'}-r(z-z')\Big)\Big|\geq\epsilon N^{\eta}\Big)$$

$$=\Pr\Big(\sup_{\bar{h}_j\in H_{j,N}^{\rho_j}}\Big|\frac{N\bar{h}_1^5\bar{h}_2}{\epsilon N^{\eta}}\sum_{z=1}^{N}\sum_{z'=1}^{N}a_{zz'}(x,\bar{h})\Big(\epsilon_{zz'}-r(z-z')\Big)\Big|\geq 1\Big) \tag{2.24a}$$

$$\leq\mathbb{E}\sum_{\bar{h}_j\in H_{j,N}^{\rho_j}}\Big|\frac{N\bar{h}_1^5\bar{h}_2}{\epsilon N^{\eta}}\sum_{z=1}^{N}\sum_{z'=1}^{N}a_{zz'}(x,\bar{h})\Big(\epsilon_{zz'}-r(z-z')\Big)\Big|^{2\alpha} \tag{2.24b}$$

$$=\sum_{\bar{h}_j\in H_{j,N}^{\rho_j}}\epsilon^{-2\alpha}N^{-2\alpha\eta}\mathbb{E}\Big|\sum_{z,z'\in I_N}N\bar{h}_1^5\bar{h}_2a_{zz'}(x,\bar{h})\Big(\epsilon_{zz'}-r(z-z')\Big)\Big|^{2\alpha} \tag{2.24c}$$

$$\leq\sum_{\bar{h}_j\in H_{j,N}^{\rho_j}}\epsilon^{-2\alpha}N^{-2\alpha\eta}C_{\alpha}\sup_{z,z'\in I_N}\Big|N\bar{h}_1^5\bar{h}_2a_{zz'}(x,\bar{h})\Big|^{2\alpha} \tag{2.24d}$$

We use the Markov inequality to get equation(2.24b) and lemma.2.3.a to get equation(2.24c).

$$\Big|a_{zz'}(x,\bar{h})\Big|\leq\frac{1}{h_1^6h_2^2}\Big(\max_{(z_1,z_2)\in I_N}K^{(2,0)}(z_1,z_2)\Big)^2\lambda(A_z)\lambda(A_{z'})=O\Big(\frac{1}{N^2h_1^6h_2^2}\Big)$$

$$\sup_{zz'\in I_N}\Big(N^2\bar{h}_1^{10}\bar{h}_2^2a_{zz'}^2(x,\bar{h})\Big)^{\alpha}=O\Big(N^2\bar{h}_1^{10}\bar{h}_2^2\frac{1}{N^4\bar{h}_1^{12}\bar{h}_2^4}\Big)=O\Big(N^{-2}\bar{h}_1^{-2}\bar{h}_2^{-2}\Big)$$

$$\sum_{\bar{h}_j \in H_{j,N}^{\rho^j}} \epsilon^{-2\alpha} N^{-2\alpha\eta} C_\alpha \sup_{zz' \in I_N} \left| N\bar{h}_1^5 \bar{h}_2 a_{zz'}(x,\bar{h}) \right|^{2\alpha} = O\Big(N^{\rho_1+\rho_2} N^{-2\alpha\eta} N^{-2} \bar{h}_1^{-2} \bar{h}_2^{-2} \Big)$$

since $O\Big(N^{\rho_1+\rho_2} N^{-2\alpha\eta}\Big) \to 0$ for $N \to \infty$ and α large enough. Then

$$\sup_{\bar{h}_j \in H_{j,N}^{\rho^j}} \left| N\bar{h}_1^5 \bar{h}_2 \sum_{zz'=1}^{N} a_{zz'}(x,\bar{h}) \Big(\epsilon_z \epsilon_{z'} - r(z-z') \Big) \right| = o_p\Big(N^{\eta-1} h_1^{-1} h_2^{-1} \Big)$$

$$\mathcal{V}_2(x,h) = o_p\Big(\frac{1}{N h_1^5 h_2} N^{\eta-1} h_1^{-1} h_2^{-1} \Big) = o_p\Big(N^{\eta-2} h_1^{-6} h_2^{-2} \Big) = o_p\Big(N^{\eta-2} h_1^{-8} \Big) \tag{2.25}$$

Treatment of $\mathcal{M}_1(x,h)$ is analogously to $\mathcal{V}_2(x,h)$.

$$\begin{aligned} \mathcal{M}_1(x,h) &= 2 \sum_{z=1}^{N} \sum_{z'=1}^{N} \int_{A_z} \int_{A_{z'}} K_h^{(2,0)}(x-u) K_h^{(2,0)}(x-v) \Big(m(x_z) - m(u) \Big) du dv \epsilon_{z'} \\ &= \sum_{z'} d_{z'}(x,h) \epsilon_{z'} \end{aligned}$$

where

$$d_{z'}(x,h) = 2 \int_{A_{z'}} \sum_{z'=1}^{N} \int_{A_z} K_h^{(2,0)}(x-u) K_h^{(2,0)}(x-v) \Big(m(x_z) - m(u) \Big) du dv$$

and $\Big| d_{z'}(x,h) \Big| = O\Big(\frac{1}{h_1^5 h_2 n^3} \Big)$

We use lemma.2.3b. for getting

$$\sup_{z' \in I_N} \Big(d_{z'}(x,h) \Big)^2 = O\Big(h_1^{-10} h_2^{-2} N^{-3} \Big)$$

$$\mathcal{M}_1(x,h) = o_p\Big(N\eta - \frac{3}{2} h_1^{-5} h_2^{-1} \Big) = o_p\Big(N^{\eta - \frac{3}{2}} h_1^{-6} \Big) \tag{2.26}$$

$$\mathcal{B}^2(x,h) = \Big[m^{(2,0)}(x,y)\Big]^2 + O\Big(\tilde{h}_1^2 + \tilde{h}_1^{-2}N^{-\frac{1}{2}} + \tilde{h}_1^{-4}N^{-1}\Big) + o_p\Big(N^{-\gamma}(\tilde{h}_1^2 + \tilde{h}_1^{-2}N^{-\frac{1}{2}} + \tilde{h}_1^{-4}N^{-1} + \tilde{h}_1^{-4}N^{-1-\gamma})\Big) \tag{2.27}$$

For the complete calculation of $\mathcal{B}^2(x,h)$, look at [30].

The last part is the investigation of $\mathcal{M}_2(x,h)$. We split $\mathcal{M}_2$ into

$$\mathcal{M}_2(x,\tilde{h}) + [\mathcal{M}_2(x,h) - \mathcal{M}_2(x,\tilde{h})]$$

with $h = \tilde{h}(1 + o_p(N^{\gamma}))$

$$\begin{aligned}\mathcal{M}_2(x,\tilde{h}) &= 2\sum_{z=1}^{N}\sum_{z'=1}^{N}\int_{A_z}\int_{A_{z'}} K_{\tilde{h}}^{(2,0)}(x-u)K_{\tilde{h}}^{(2,0)}(x-v)m(u)du\ dv\epsilon_{z'} \\ &= \sum_{z'=1}^{N}\mathcal{C}_{z'}(x,\tilde{h})\epsilon_{z'}\end{aligned}$$

where

$$\begin{aligned}\mathcal{C}_{z'} &= \frac{2}{\tilde{h}_1^6\tilde{h}_2^2}\int_{A_{z'}}\sum_{z=1}^{N}\int_{A_z} K^{(2,0)}\Big(\tilde{H}^{-1}(x-u)\Big)K^{(2,0)}\Big(\tilde{H}^{-1}(x-v)\Big)m(v)du\ dv \\ &= \frac{2}{\tilde{h}_1^6\tilde{h}_2^2}\int_{A_{z'}} K^{(2,0)}\Big(\tilde{H}^{-1}(x-u)\Big)du\ S(x,\tilde{h})\end{aligned} \tag{2.28}$$

with

$$\begin{aligned}S(x,\tilde{h}) &= \frac{1}{\tilde{h}_1^3\tilde{h}_2}\int_A K^{(2,0)}\Big(\tilde{H}^{-1}(x-u)\Big)du \\ &= \frac{1}{\tilde{h}_1^2}\int_T K^{(2,0)}(z_1,z_2)m\left(\begin{pmatrix}x\\y\end{pmatrix} - \tilde{H}\begin{pmatrix}z_1\\z_2\end{pmatrix}\right) \qquad (2.29)\\ &= m^{(2,0)}(x) + O(\tilde{h}_1^2 + \tilde{h}_2^2) \qquad (2.30)\end{aligned}$$

With $|\mathcal{C}_z(x,\tilde{h})| = O\Big(\frac{1}{N\tilde{h}_1^{-3}\tilde{h}_2}\Big)$, $\sup_{z\in I_N}\mathcal{C}_z^2(x,\tilde{h}) = O(N^{-2}\tilde{h}_1^{-6}\tilde{h}_2^{-2})$ and using a similar calculation as for $\mathcal{M}_1(x,h)$ we get

$$\mathcal{M}_2(x,\tilde{h}) = o_p(N^{-1+\eta}\tilde{h}_1^{-3}\tilde{h}_2^{-1}) = o_p(N^{-1+\eta}\tilde{h}_1^{-4}) \tag{2.31}$$

$$\mathcal{M}_2(x,y;h_1,h_2)-\mathcal{M}_2(x,y;\tilde{h}_1,\tilde{h}_2)=\sum_{z=1}^{N}\Big(\mathcal{C}_z(x,y,h_1,h_2)-\mathcal{C}_z(x,y,\tilde{h}_1,\tilde{h}_2)\Big)\epsilon_z$$
$$=\sum_{z=1}^{N}\Big(\mathcal{C}_z(x,y,h_1,h_2)-\mathcal{C}_z(x,y,\tilde{h}_1,h_2)\Big)\epsilon_z \quad (2.32a)$$
$$+\sum_{z=1}^{N}\Big(\mathcal{C}_z(x,y,\tilde{h}_1,h_2)-\mathcal{C}_z(x,y,\tilde{h}_1,\tilde{h}_2)\Big)\epsilon_z \quad (2.32b)$$

$$|\mathcal{C}_z(x,y;h_1,h_2)-\mathcal{C}_z(x,y;\tilde{h}_1,h_2)|\le\frac{C|\tilde{h}_1-h_2|}{Nh_1^3h_2h_1^2}\{\tilde{h}_1^3+h_2^3+(\tilde{h}_1^2+h_2^2)h_1\}+\frac{C|h_1^3-h_1^3|}{Nh_1^3h_2\tilde{h}_1^3}$$
$$\le C|\tilde{h}_1-h_1|T_1(h_1)$$

where $T_1(h_1):=\frac{\tilde{h}_1^3+h_1^2}{Nh_1^3h_2^2\tilde{h}_1^3}+\frac{1}{Nh_1^3h_2h_1^2}\{\tilde{h}_1^3+h_2^3+(\tilde{h}_1^2+h_2^2)h_1\}$
and

$$|\mathcal{C}_z(x,y;\tilde{h}_1,h_2)-\mathcal{C}_z(x,y;\tilde{h}_1,\tilde{h}_2)|\le\frac{C|\tilde{h}_2-h_2|}{N\tilde{h}_1^3h_2\tilde{h}_1^2}(\tilde{h}_1^3+\tilde{h}_2^3)+\frac{C|\tilde{h}_2-h_2|\tilde{h}_1^3}{N\tilde{h}_1^3h_2\tilde{h}_1^3\tilde{h}_2}$$
$$\le C|\tilde{h}_2-h_2|T_2(h_2)$$

where $T_2(h_2):=\frac{1}{N\tilde{h}_1^3\tilde{h}_2h_2}+\frac{1}{N\tilde{h}_1^3h_2\tilde{h}_1^2}\{\tilde{h}_1^3+\tilde{h}_2^3\}$
Then by (2.32a)

$$\Big|\frac{1}{T}\sum_z\Big(\mathcal{C}_z(x,y;h_1,h_2)-\mathcal{C}_z(x,y;\tilde{h}_1,h_2)\Big)\epsilon_z\Big|$$
$$\le\sup_{\bar{h}_1\in H_{1,N}^{\rho_1}}\Big|\frac{1}{\overline{T}}\sum_z\Big(\mathcal{C}_z(x,y;\bar{h}_1,h_2)-\mathcal{C}_z(x,y;\tilde{h}_1,h_2)\Big)\epsilon_z\Big| \quad (2.33a)$$
$$+\inf_{\bar{h}_1\in H_{1,N}^{\rho_1}}\Big|\frac{1}{T}\sum_z\Big(\mathcal{C}_z(x,y;h_1,h_2)-\mathcal{C}_z(x,y;\tilde{h}_1,h_2)\Big)\epsilon_z$$
$$-\frac{1}{\overline{T}}\sum_z\Big(\mathcal{C}_z(x,y;\bar{h}_1,h_2)-\mathcal{C}_z(x,y;\tilde{h}_1,h_2)\Big)\epsilon_z\Big| \quad (2.33b)$$

where $T := \sqrt{Nh_2(\tilde{h}_1 + h_1)}\ |\tilde{h}_1 - h_1|\ T_1(h_1)$ and $\bar{T} := \sqrt{Nh_2(\tilde{h}_1 + \bar{h}_1)}\ |\tilde{h}_1 - h_1|\ T_1(\bar{h}_1)$, so that,
$C = \sqrt{Nh_2(\tilde{h}_1 + h_1)}$ and $\bar{C} = \sqrt{Nh_2(\tilde{h}_1 + \bar{h}_1)}$.
The number of nonzero $C_z(x, y, h_1, h_2)$ and $C_z(x, y, \tilde{h}_1, h_2)$ is $O(Nh_1h_2 + N\tilde{h}_1h_2)$. Now consider equation (2.33a)

$$\Big|\mathcal{C}_z(x, y; h_1, h_2) - \mathcal{C}_z(x, y; \tilde{h}_1, h_2)\Big| \leq C|\tilde{h}_1 - h_1|T_1(h_1)$$

$\Big(\frac{1}{T}\Big)^2 \sum_z \Big(\mathcal{C}_z(x, y; h_1, h_2) - \mathcal{C}_z(x, y; \tilde{h}_1, h_2)\Big)^2 = O\Big(\frac{1}{Nh_2(h_1+\tilde{h}_1)} \frac{(|\tilde{h}_1-h_1|T_1(h_1))^2}{(|\tilde{h}_1-h_1|T_1(h_1))^2}\Big) = O\Big(\frac{1}{Nh_2(h_1+\tilde{h}_1)}\Big)$
and we get that (2.33a) is $o_p(N^{\eta-\frac{1}{2}})$

Consider equation (2.33b), it has to be shown that

$$\frac{\mathcal{C}_z(x, y; h_1, h_2) - \mathcal{C}_z(x, y; \tilde{h}_1, h_2)}{|\tilde{h}_1 - h_1|} \tag{2.34}$$

is Lipschitz continuous as a function of $h_1, \frac{1}{T}$ is also Lipschitz continuous, because the denominator is bounded away from zero, look at [30], so that, equation(2.33b) is $o_p(n^{\rho_1} + n^{\rho_2}) = o_p(1)$ if ρ_1 and ρ_2 are chosen large enough.

We get, equation(2.32a) will be equal to

$$o_p(TN^{\gamma-\frac{1}{2}}h_2^{-\frac{1}{2}}(h_1 + \tilde{h}_1)^{-\frac{1}{2}}) = o_p(N^{\eta-\gamma-1}h_2^{-\frac{1}{2}}\tilde{h}_2^{-\frac{1}{2}}\tilde{h}_1^{-\frac{5}{2}}(h_1 + \tilde{h}_1)^{-\frac{1}{2}})$$

The treatment of (2.32b) is similar to (2.32a) and we get

$$o_p(N^{\eta-\gamma-1}h_2^{-\frac{1}{2}}\tilde{h}_2^{-\frac{1}{2}}\tilde{h}_1^{-\frac{5}{2}}(h_1 + \tilde{h}_1)^{-\frac{1}{2}})$$

Hence

$$\mathcal{M}_2(x, h) = o_p(N^{\eta-1}\tilde{h}_1^{-4}) \tag{2.35}$$

Summing up we get

$$\begin{aligned}[\hat{m}^{(2,0)}(x, h)]^2 = [m^{(2,0)}(x, y)]^2 &+ O(\tilde{h}_1^2 + \tilde{h}_1^{-2}N^{-\frac{1}{2}} + \tilde{h}_1^{-4}N^{-1}) \\ &+ o_p(N^{-\gamma}(\tilde{h}_1^2 + \tilde{h}_1^{-2}N^{-\frac{1}{2}} + \tilde{h}_1^{-4}N^{-1} + \tilde{h}_1^{-4}N^{-1-\gamma})) \\ &+ o_p(N^{-\frac{1}{2}+\eta}\tilde{h}_1^{-4}) + o_p(N^{\eta-2}h_1^{-8}) + \frac{f(0,0)}{N^2h_1^3h_2}Q_k^{(2,0)} + O(\frac{1}{N^4h_1^6h_2^2})\end{aligned} \tag{2.36}$$

Further steps are are not deviate from [30] and they can be found in p.50-57. □

Remark 2.2. *In contrast to the analogous one-dimensional Theorem of [47] , bias and standard error of $\hat{f}_M(0,0)$ do not show up in the difference between $\hat{h}^{(7)}$ and h_{opt} as, by our choice of M and by Corollary 2.1, they are of order $o(n^{-\frac{1}{3}})$ and, therefore, asymptotically negligible.*

Theorem 2.2. *Under the assumption of Theorem2.1*
a) $\mathbb{E}\hat{h}^{(7)} - h_{IASY} = O(n^{-\frac{2}{3}})$
b) $n^{-\frac{4}{3}}\big(\hat{h}^{(7)} - \mathbb{E}\hat{h}^{(7)}\big)$ *converges in distribution to a bivariate normal distribution with mean 0, if* $\int_A \big\{\frac{\partial^4}{\partial x_1^4} m(x)\big\}^2 dx$, $\int_A \big\{\frac{\partial^4}{\partial x_2^4} m(x)\big\}^2 dx > 0$

2.8 Simulation

The design of this simulation will follow (2.1) with regression model in (2.2). In this simulation study we are going to compare three methods of smoothing:

i) pretending uncorrelatedness of the residuals ϵ_{ij}, where the spectral density is constant and equal to $\sigma^2 = Var\ \epsilon_{ij}$, using only two bandwidth parameters,

ii) allowing correlated ϵ_{ij}, but using only two bandwidth parameters h_1, h_2, i.e. we adapt the degree of smoothing only along coordinate directions,

iii) the general case with correlated ϵ_{ij} and three bandwidth parameters.

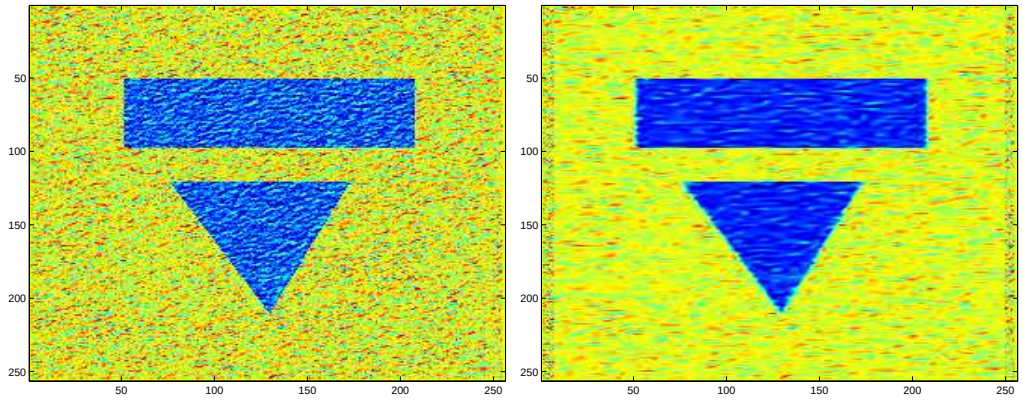
We use (2.7) to estimate the autocovariance, and for the uncorrelated variance, the estimator is given as follow

$$\hat{\sigma}^2 = \frac{4}{9(n-2)(n-2)} \sum_{i=2}^{n-1} \sum_{j=2}^{n-1} \hat{\epsilon}_{ij}^2$$

where

$$\begin{aligned}\hat{\epsilon}_{ij}^2 =& Y_{ij} - \frac{1}{2}\Big(Y_{i-1,j} + Y_{i+1,j} + Y_{i,j-1} + Y_{i,j+1}\Big) \\ &+ \frac{1}{4}\Big(Y_{i-1,j-1} + Y_{i-1,j+1} + Y_{i+1,j-1} + Y_{i+1,j+1}\Big)\end{aligned}$$

If m is twice continuously differentiable, then the bias of $\hat{\sigma}^2$ is $O(n^{-1})$ as in [48]



The kernel $K_{0,0}(x, y)$ is the two-dimensional Epanechnikov kernel

$$K(x, y) = \frac{2}{\pi}(1 - (x^2 + y^2))^2$$

The kernel $K_{1,1}(x, y), K_{2,0}(x, y), K_{0,2}(x, y)$ are derived by differentiating

$$\hat{K}(x, y) = \frac{3}{\pi}(1 - (x^2 + y^2))^2,$$

so

$$K_{1,1}(x, y) = \frac{24}{\pi}xy,$$
$$K_{2,0}(x, y) = \frac{12}{\pi}(-1 + 3x^2 + y^2)$$

and

$$K_{0,2}(x, y) = \frac{12}{\pi}(-1 + x^2 + 3y^2)$$

with all support in the unit ball.

The noise in the image is generated correlatedly by

$$\begin{aligned}\epsilon_{ij} = &-0.5\eta_{i,j-1} + 0.25\eta_{i,j+1} - 0.5\eta_{i-1,j} + 0.25\eta_{i+1,j} \\ &- 0.5\eta_{i-1,j-1} + 0.25\eta_{i+1,j-1} - 0.5\eta_{i-1,j+1} + 0.25\eta_{i+1,j+1}\end{aligned}$$

where $\eta \sim \mathcal{N}(0, 1)$

We can see clearly in this simulation result below, that the "uncorrelated" smoother, could not denoise those images as well as the other two methods. However, between

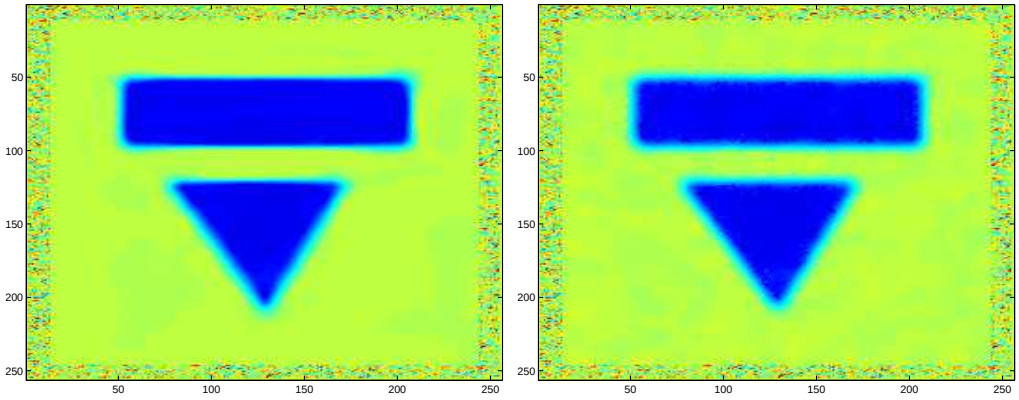


Figure 2.1: From top left to bottom right: the noise image, smoothed with uncorrelated variance and 2 parameters bandwidth, smoothed with correlated variance and 2 parameters bandwidth, smoothed with correlated variance and 3 parameters bandwidth

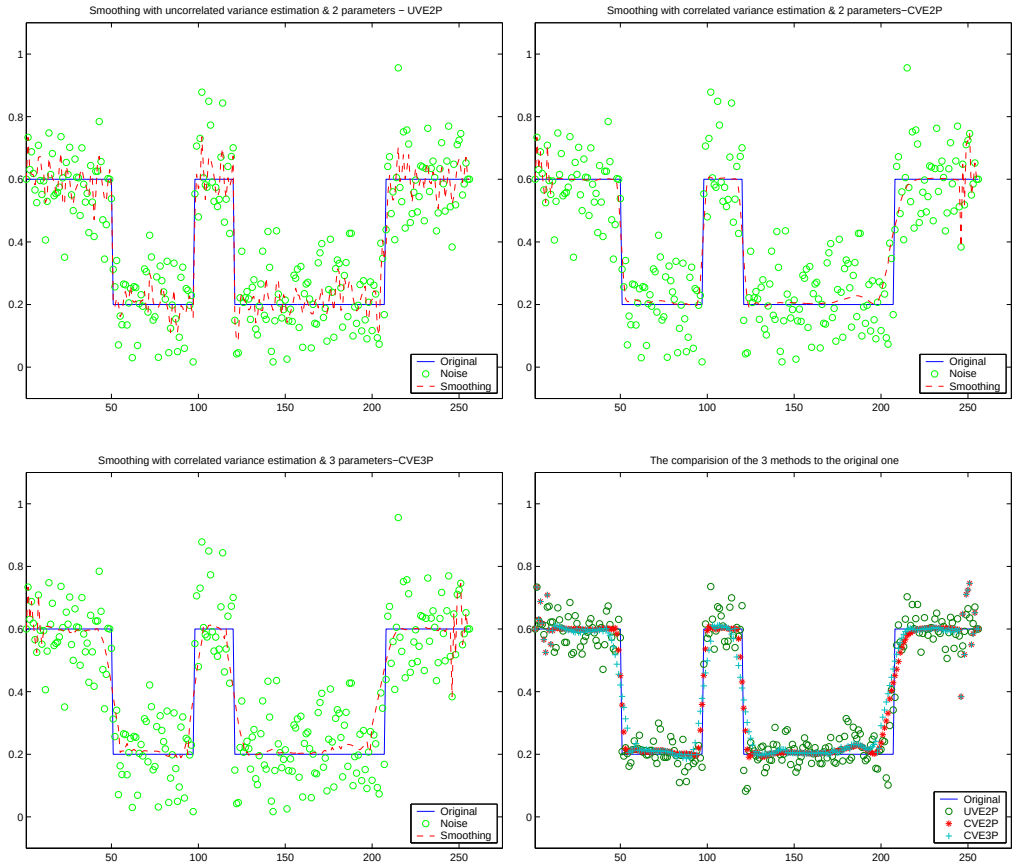


Figure 2.2: A vertical section through point 128, showing the noisy data (dot), the true curve (line) and the estimated curve (dash) for image in fig(2.1), respectively, and the comparison with them

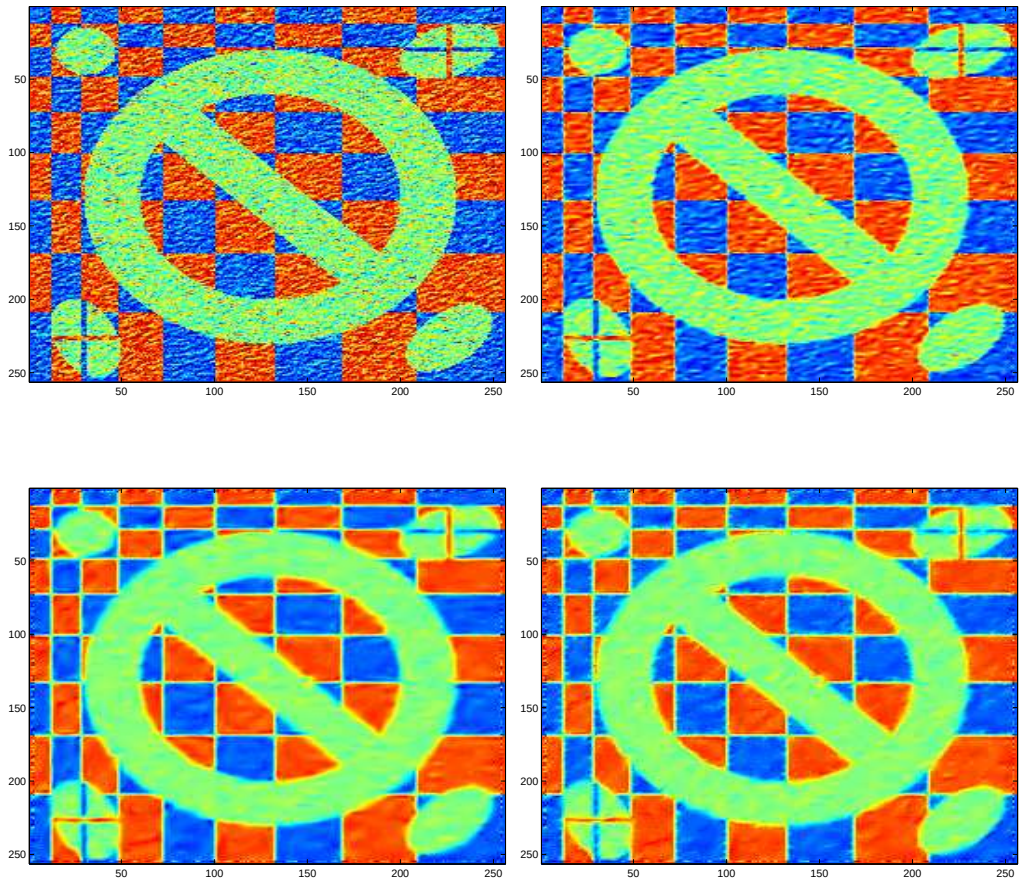
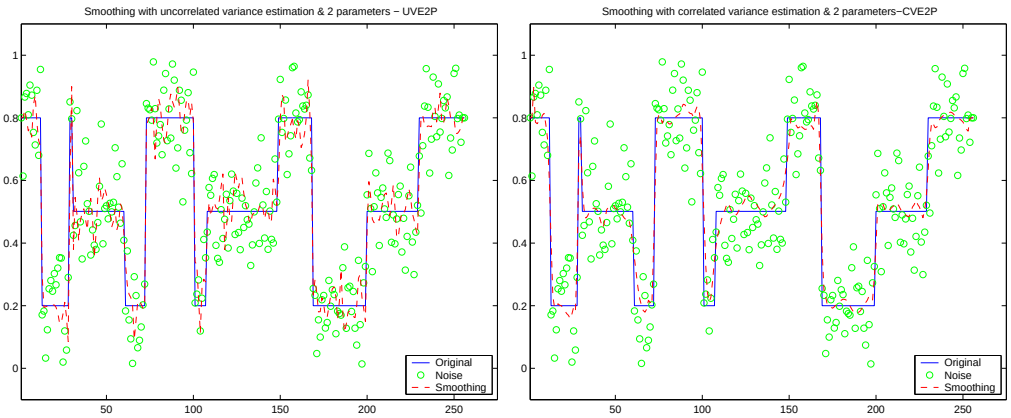


Figure 2.3: From top left to bottom right: the noise image, smoothed with uncorrelated variance and 2 parameters bandwidth, smoothed with correlated variance and 2 parameters bandwidth, smoothed with correlated variance and 3 parameters bandwidth



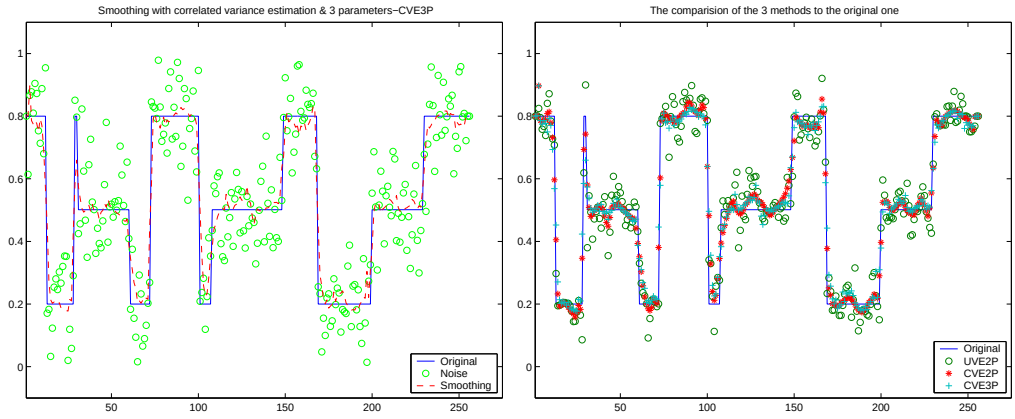


Figure 2.4: A vertical section through point 128, showing the noisy data (dot), the true curve (line) and the estimated curve (dash) for image in fig(2.3), respectively, and the comparison with them

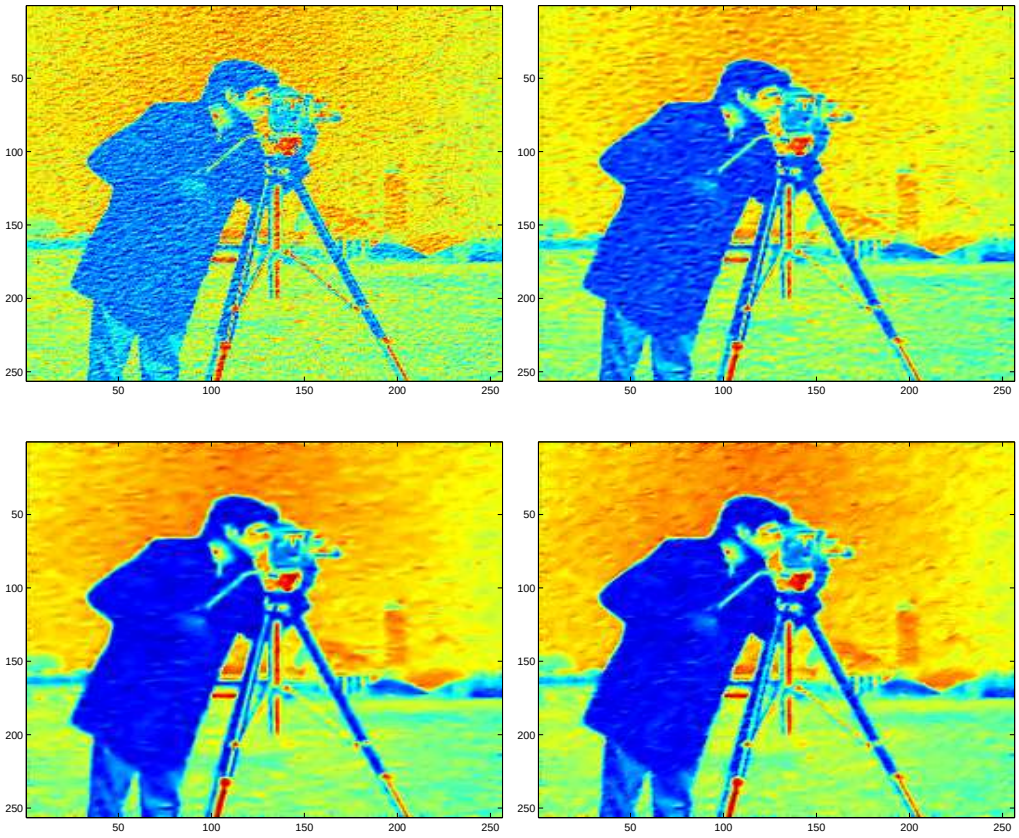


Figure 2.5: From top left to bottom right: the noise image, smoothed with uncorrelated variance and 2 parameters bandwidth, smoothed with correlated variance and 2 parameters bandwidth, smoothed with correlated variance and 3 parameters bandwidth

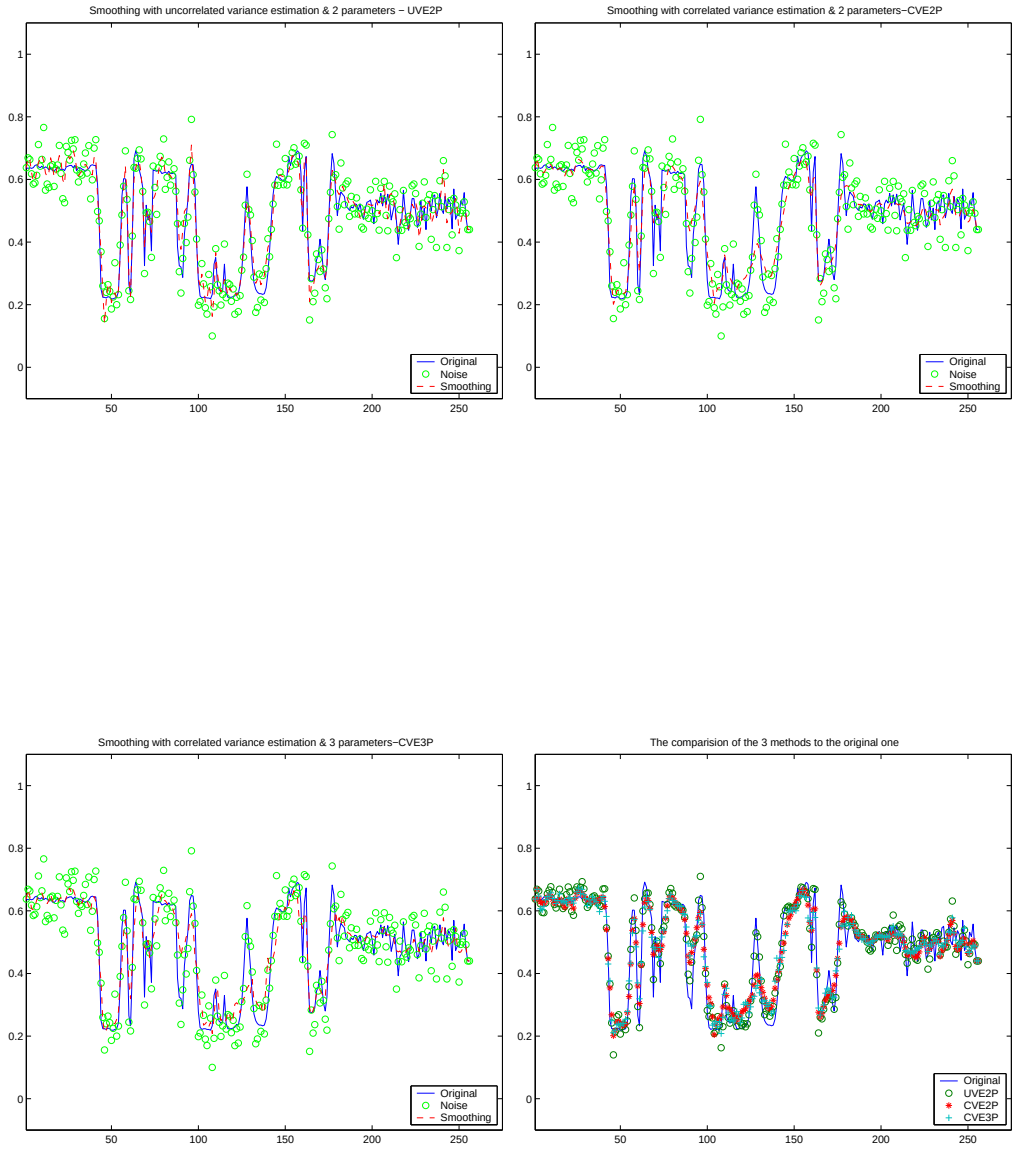


Figure 2.6: A vertical section through point 128, showing the noisy data (dot), the true curve (line) and the estimated curve (dash) for image in fig(2.5), respectively, and the comparison with them

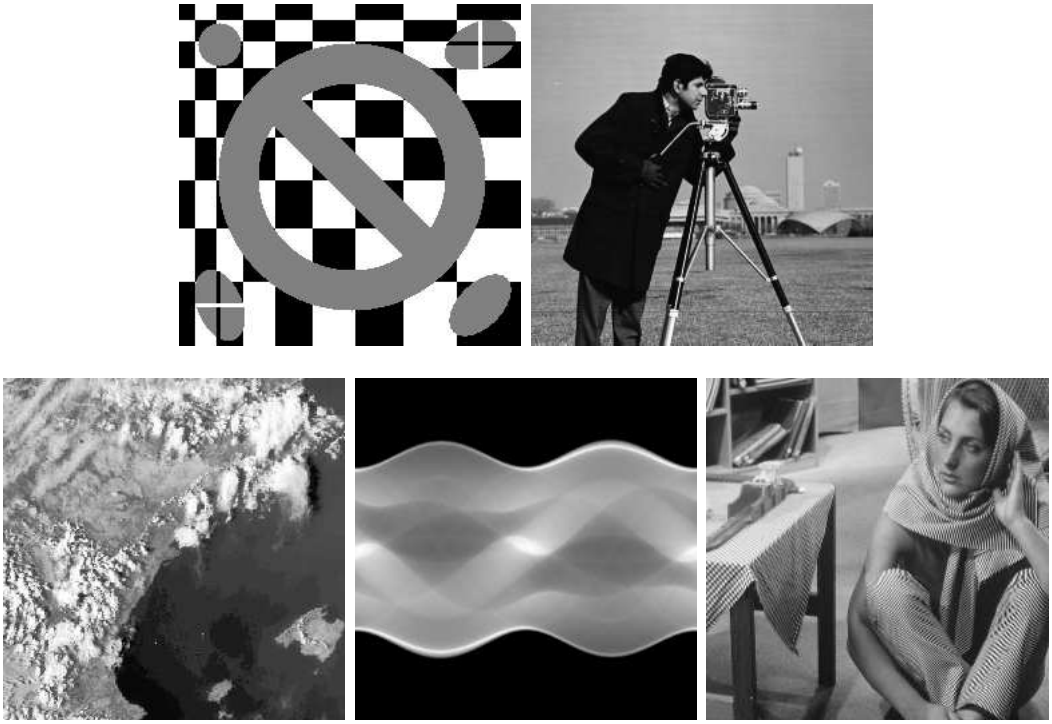


Figure 2.7: DFG-SPP 1114 Test Images: chess board & circle, camera, satellite, radon, barbara

the latter two methods, even though three parameters of bandwidth seems to be more sophisticated and needs running time longer than in the two parameters case, it could not give a significant improvement. We suspect that our procedure to transform the indefiniteness away does not give an optimal result yet. Therefore we suggest a further investigation in that direction.

2.8.1 Image denoising test

In this following section we are going to give a summary result of image denoising as a part of an interdisciplinary evaluation of mathematical methods in signal and image processing in the DFG-SPP 1114 project. The data are provided from the DFG-SPP 1114 homepage [75]. It consists of 5 images, which are depicted in the Figures (2.7).

For the calculations the gray values of these images were scaled to the interval [0,1]. The noisy images are generated as white noise with standard deviation 0.25, 0.5 and 1, respectively, which has been added to a locally constant test image by the following

process:

1. Scale the gray values to the interval [0.2, 0.8].
2. Add white noise of different variances.
3. Crop the values back to the interval [0,1], i. e. values below zero or above one are set to zero resp. one.

This type of noise may cause a little trouble, since this approach will introduce biases, but it is standard in image processing.

The evaluation criteria should show that a good noise reduction yields a small difference between the denoising result $\hat{Y}$ and the original function Y. In this test, there will be 80 realizations for each image and each different standard deviation respectively. The summary of the test will be the mean values of that data for every different norms, given as follows.

Let Y be the original image, and $\hat{Y}$ the denoised image. Both are of size n_x times n_y and rescaled to have gray values in $[0, 1]$.

- the L_1 - norm
$$\|Y - \hat{Y}\|_{L_1} = \frac{1}{n_x n_y} \sum_{i}^{n_x} \sum_{j}^{n_y} |Y(i,j) - \hat{Y}(i,j)|$$
- the L_2 - norm or mean square integrated error
$$\|Y - \hat{Y}\|_{L_2}^2 = \frac{1}{n_x n_y} \sum_{i}^{n_x} \sum_{j}^{n_y} |Y(i,j) - \hat{Y}(i,j)|^2$$
- the L_∞ - norm
$$\|Y - \hat{Y}\|_{L_\infty} = \max_{1 \le i \le n_x, 1 \le j \le n_y} |Y(i,j) - \hat{Y}(i,j)|^2$$
- Large Deviation Probability (LDP), i.e., the percentage of points, were the estimate differs from the true value by more than the minimal contrast divided by four (for which this evaluation is 0.15))
$$LDP(Y, \hat{Y}) = \frac{1}{n_x n_y} \#\{(i,j) \big| \ \ |Y(i,j) - \hat{Y}(i,j)| > 0.15\}$$

- The Peak Signal to Noise Ratio (PSNR)

$$PSNR(Y, \hat{Y}) = 20 \log_{10} \Big(\frac{0.6}{\|Y - \hat{Y}\|_{L^2}} \Big)$$

A good denoising procedure will give a small error in the sense of L_1, L_2, L_∞ and LDP, but show a strong peak signal to noise ratio measurement.

Remark:.

Marron and Tsybakov in [63] pointed out that this classical mathematical norms on function space can be inappropriate from a graphical viewpoint, since these norms are all based on the vertical distances between curves. The human eyes use both horizontal and vertical information. Marron and Tsybakov stated that a mathematical method for capturing this is to treat the curves not as functions of a single variable but instead as sets of points in the plane. For example, a given continuous function $f : [a, b] \to \mathbb{R}$ can be represented by its "graph"

$$G_f = \{(x, y) : x \in [a, b], y = f(x)\} \subset \mathbb{R}^2.$$

Then the distance between sets can be formulated, e.g., as Hausdorff distance as follows

$$d((x, y), G) = \inf_{(x',y') \in G} \|(x, y) - (x', y')\|_2;$$

that is, the shortest distance from the given point (x,y) to any point in the closed set G, where $\|.\|_2$ denotes the euclidean distance. Distances from the points in the set G_1 to the set G_2 can then be combined into the set distances

$$\mathcal{D}(G_1, G_2) = \{d((x, y), G_2) : (x, y) \in G_1\}$$

To measure the asymmetric "visual error" criteria, they proposed a class of summaries of $\mathcal{D}(G_1, G_2)$, as

$$VE_i(f_1 \to f_2) = \Big[\int_a^b d((x, f_1(x)), G_{f_2})^i dx \Big]^{1/i}$$

where $i = 1, 2, \infty$ (replacing the integral by the sup norm for $i = \infty$

and the symmetric error criterias are proposed as

$$\begin{aligned} SE_1(f_1, f_2) &= VE_1(f_1 \to f_2) + VE_1(f_2 \to f_1) \\ SE_2(f_1, f_2) &= [VE_2(f_1 \to f_2)^2 + VE_2(f_2 \to f_1)^2]^{1/2} \\ SE_\infty(f_1, f_2) &= d_H(G_{f_1}, G_{f_2}) \\ &= \max(VE_\infty(f_1 \to f_2), VE_\infty(f_2 \to f_1) \end{aligned}$$

where

$$d_H(G_1, G_2) = max\{sup(\mathcal{D}(G_1, G_2)), sup(\mathcal{D}(G_2, G_1))\}$$

is a Hausdorff distance.

In common practice of constructing curves for plotting on an equally spaced, with gird spacing Δx, on a compact grid of x locations $\mathcal{X}$, the discretized version of graph G_f of function $f(x)$ is denoted by $g_f^{discr} = \{(x, f(x) : x \in \mathcal{X}\}$. Then the discretized version of $VE_i(f_1 \to f_2)$ is given by

$$VE_i^{discr}(f_1 \to f_2) = \left[\Delta x \sum_{x \in \mathcal{X}} d(x, f_1(x)), G_2^{discr})^i\right]^{1/i}$$

■

Table(2.1) was taken from [75]. It gives the evaluation of image denoising using four standard tools and the error measures for the non-processed noisy data as a kind of benchmark

1. Evaluation of the noisy data, i.e. without any anterior denoising (ND)
2. Simple Median Filter (MF)
3. Simple Average Filter(AF)
4. Simple Gaussian Filter (GF)
5. Wavelet Denoising realized with the Wavelet Toolbox of MATLAB 6
   (a) MWH - Matlab Wavelet (Haar wavelet, hard threshold)
   (b) MWCx - Matlab Wavelet (coifx wavelet, x = 3,4,5; soft treshhold)

Table 2.1: Summary of Denoising of DFG SPP 1114's Images Test Simple Methods as an Comparison

Image	σ	Method	Mask	L_1	L_2	L_∞	LDP	PSNR
		ND		0.135830	0.041172	0.994760	0.218550	13.85410
		MF	3x3	0.068648	0.013247	1.000000	0.053055	18.79570
	0.25	AF	3x3	0.113340	0.022129	0.672370	0.092804	16.51880
		GF	5x5	0.117560	0.022950	0.682280	0.109680	16.36010
		MWH		0.092641	0.012659	0.739460	0.026443	18.94920
		ND		0.240880	0.120410	1.000000	0.425400	9.193500
Chess Board		MF	5x5	0.112820	0.029973	1.000000	0.130390	15.24810
&	0.50	AF	3x3	0.178050	0.046426	0.814000	0.244290	13.34340
Circle		GF	5x5	0.203740	0.066621	0.897090	0.330230	11.77010
		MWH		0.158820	0.034721	0.828540	0.123030	14.60930
		ND		0.348370	0.230340	1.000000	0.553220	6.376500
		MF	7x7	0.165460	0.059280	1.000000	0.256180	12.28570
	1.00	AF	5x5	0.242440	0.084016	0.775890	0.521330	10.72100
		GF	5x5	0.294640	0.132950	0.982450	0.488450	8.734000
		MWH		0.229340	0.075251	0.857430	0.532810	11.22260

We only state tables for the chess board and circle image, the complete table can be looked at in Appendix A.

The second table we are going to analyse is Table (2.2). This table shows the result for denoising the same image using kernel based method discussed in the beginning of this section. The three different procedures that we use can be abbreviated in this following manner

1. CV2P - Constant Variance estimator and 2 parameters of bandwidth
2. AC2P - Autocovariance estimator and 2 parameters of bandwidth
3. AC3P - Autocovariance estimator and 3 parameters of bandwidth

By an instant glimpse at these two tables, we can conclude that the kernel based methods give more superior results than the standard tools in all aspects of measurements.

Table 2.2: Summary of Denoising of DFG SPP 1114's Images Test using Kernel Based Methods

Images	σ	Method	L_1	L_2	L_∞	LDP	PSNR
Chess Board & Circle	0.25	CV2P	0.0226	0.0021	0.3537	0.0166	22.3414
		AC2P	0.0245	0.0025	0.3588	0.0250	21.5323
		AC3P	0.0282	0.0035	0.3747	0.0742	20.1557
	0.50	CV2P	0.0308	0.0027	0.3754	0.0301	21.2213
		AC2P	0.0317	0.0033	0.3841	0.0469	20.4257
		AC3P	0.0351	0.0038	0.3780	0.0666	19.7913
	1.00	CV2P	0.0591	0.0088	0.7201	0.1253	16.1309
		AC2P	0.0484	0.0059	0.4282	0.0861	17.8785
		AC3P	0.0521	0.0069	0.5556	0.0923	17.1664

The last table was taken from Polzehl and Spokoiny([72]). They developed a nonparametric estimation procedure so called Adaptive Weight Smoothing(AWS) and compared it to the five other methods.

Table 2.3: The denoising quality evaluated by two different error measures MISE and LDP for Chess Board & Circle's image

Methods	MISE $\sigma = 0.25$	MISE $\sigma = 0.5$	MISE $\sigma = 1$	LDP $\sigma = 0.25$	LDP $\sigma = 0.5$	LDP $\sigma = 1$
AWS	0.0021	0.0109	0.0328	0.007	0.032	0.119
Gauss filtering	0.0138	0.0243	0.0396	0.212	0.313	0.452
Nonlinear Gauss	0.0096	0.0262	0.0454	0.151	0.334	0.491
Modal regression	0.0068	0.0254	0.0426	0.078	0.290	0.479
wavethresh	0.0079	0.0147	0.0475	0.073	0.172	0.437
Markov Random Fields	0.0050	0.0204	0.0475	0.048	0.248	0.497

In this particular image, where there are many sudden changes in the gray level and otherwise, the gray level is locally constant, comparing with AWS, our new methods cannot give an improvement particularly for small noise, i.e., $\sigma = 0.25$. However, it gives a significant improvement when the noise is big enough. This can be understood, since

if the noise is generated independently and too small, this causes estimating the noise contribution by an autocovariance estimator which seems to be unnecessary. To the other four methods the kernel based procedures demonstrate a better performance.

Chapter 3

Texture modeling using Markov Random Fields

Texture modeling can be based on deterministic models or on probabilistic models. The deterministic models usually rely on heuristics and end up with a new algorithm, e.g., Heeger and Bergen [46], Efros et.al.([25], [24]), Li Yi Wei [85]. They give good visual results but have no theoretical justification. On the other hand the probabilistic models, mainly based on random field, e.g.,Besag,([6], [7]),Geman and Geman [33],Paget [68], Winkler [87], and spatial statistics, e.g. Kashyap and Chellappa ([53],[12]). They give good results, particularly for random textures, however it is hard to model a texture using random fields if it has a regularity characteristic, such as fabric texture as in the *Brodatz picture gallery*,[8].

Therefore starting from the random field based approach we are going to construct a model which has randomness structure as well as regularity properties, i.e., semi regular texture.

3.1 Markov Random Fields

To model a digital image as a realization of a Markov Random Field (MRF), we consider a finite index set $S = \{1, 2, \ldots n\}$ to be given and the elements $s \in S$ denote the *pixels* or *sites* of an image x. Assume that a finite set of discrete labels or gray scale value $\mathcal{L}_s \subseteq \mathcal{L}$ is given for every site $s \in S$. The site value x_s is then contained in the *state space*

$\mathcal{L} = \{0, \ldots, L-1\}$, where L is the number of gray levels. The maximum L considered here is 256. Now, an image can be viewed as an element of the finite product space χ which is defined as the product of all label set $\mathcal{L}_s$,i.e.

$$\chi = \prod_{s \in S} \mathcal{L}_s$$

We call χ the *configuration space* and an element $x \in \chi$ a configuration.

Let Π be a strictly positive (joint) probability measure on χ with $\Pi(x) > 0, \forall x \in \chi$. We call this probability measure a *Random Field*.

Besag [6] proved that the joint distribution $\Pi(x)$ is uniquely determined by its local conditional probability density functions, $\Pi_s(x_s|x_r, r \neq s)$. Those sites which possibly influence the local characteristic at a site s will be called neighbors of s. The neighborhood relation fulfills the following axioms.

Definition 3.1. *Neighborhood System*
Let $\mathcal{N}$ denote a collection of subsets of S of the form

$$\mathcal{N} := \{\mathcal{N}(s) \subset S | s \in S\}$$

$\mathcal{N}$ is called a neighborhood system for S if it satisfies

(i) $s \notin \mathcal{N}(s) \quad \forall s \in S$

(ii) $\forall s, t \in S$ *with* $s \neq t$ *we have* $s \in \mathcal{N}(t) \iff t \in \mathcal{N}(s)$

The sites $t \in \mathcal{N}(s)$ are called neighbors of s. We write $< s, t >$ if s and t are neighbors of each other.

In the case of a regular pixel grid the set of sites S can be considered to be a subset of $\mathbb{Z}^2$, e.g., $S = \{(i, j) \in \mathbb{Z}^2 | -n \leq i, j \leq n\}$ for a square lattice of size $(2n+1)$x(2n + 1). Usually we define the neighbors of a pixel (i, j) on such a grid to be the set of nearby sites whose midpoints lie within a circle for radius c, i.e.,

$$\mathcal{N}(i, j) := \{(k, l) \in S | 0 < (k-i)^2 + (l-j)^2 \leq c^2\}$$

To describe the kind of neighborhood, we define the *order* of a neighborhood with increasing c. The *first order* neighborhood system for which $c^2 = 1$ considers the four nearest pixels of the site (i, j) to be neighbors,i.e.,$(i, j+1), (i, j-1), (i+1, j), (i-1, j) \in \mathcal{N}(i, j)$. The *second order* neighborhood system for which $c^2 = 2$ considers the eight nearest pixels to be neighbors, i.e., $(i, j+1), (i, j-1), (i+1, j), (i-1, j), (i-1, j+1), (i-1, j-1), (i+1, j+1), (i+1, j-1) \in \mathcal{N}(i, j)$. , as we can see in Figure3.1, for the third order there will be twelve neareast pixels to be neighbors, and twenty neighbors for the fourth order, and so on. We will denote these neighborhood systems by $\mathcal{N}^o$, where $o \in 1, 2, ...$ corresponds to the order.

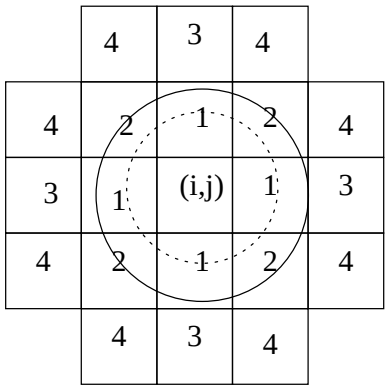


Figure 3.1: Illustration of the neighborhood systems of order 1 to 4: The pixels having their midpoints within the smaller circle are first order ($c = 1$), pixels within the larger circle are second order neighbors ($c = \sqrt{2}$).

This system can be decomposed into subsets called a *clique*, which is going to be used in the next section, in this way.

Definition 3.2. *Clique*
A subset $C \subset S$ is called a clique with respect to the neighborhood system $\mathcal{N}$ if either C contains only one element or if any two different elements in C are neighbors. We denote the set of all cliques by $\mathcal{C}$.

For any neighborhood system $\mathcal{N}$ we can decompose the set of all possible cliques into subsets $C_i, i = 1, 2, ...$, which contain all cliques of cardinality i, respectively. Then we have

$$\mathcal{C} = \bigcup_i C_i$$

with $C_1 := \{\{s\}|s \in S\}$ being the set of all singletons, $C_2 := \{\{s_1, s_2\}|s_1, s_2 \in S$ are neighbors $\}$, the set of all pair-set cliques, and so on.

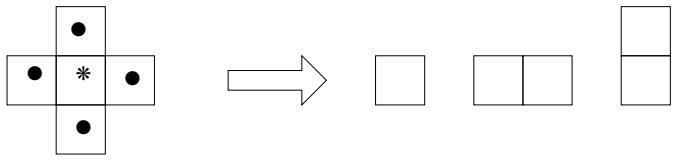


Figure 3.2: Illustration of the clique types for the neighborhood system $\mathcal{N}^1$

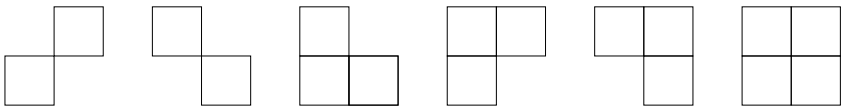


Figure 3.3: Illustration of the additional clique types for the neighborhood system $\mathcal{N}^2$

Now, we can define a Markov random field(MRF) with respect to the neighborhood system $\mathcal{N}$ by requiring

$$\Pi_s(x_s|x_r, r \neq s) = P(x_s|x_r, r \in \mathcal{N}(s)), \quad s \in S, \quad x \in \chi \tag{3.1}$$

An image is modeled by estimating this function with respect to a neighborhood system $\mathcal{N}$.

3.2 Gibbs Random Fields

To calculate the conditional probability in (3.1) we use the representation of random fields in the Gibbsian form,that is, we assume

$$\Pi(x) = \frac{exp(-H(x))}{\sum_z exp(-H(z))} = Z^{-1} exp(-H(x)) \tag{3.2}$$

which are always strictly positive and hence random fields. Such a measure Π is called the *Gibbs Random Field (GRF)* induced by the *energy function* H, and the denominator Z is called the *partition function*.

The idea and most of the terminology is borrowed from statistical mechanics, where Gibbs fields are used as models for the equilibrium states of large physical systems. Then following this terminology, it is convenient to decompose the energy into the contributions of configurations on subsets of S.

Definition 3.3. *Potential Function*
A potential is a family $\{U_A : A \subset S\}$ *of functions on* χ *such that:*

(i) $U_\emptyset(x) = 0$

(ii) $U_A(x) = U_A(y)$ *if* $x_s = y_s$ *for each* $s \in A$

The energy of the potential U *is given by*

$$H_U := \sum_{A \subset S} U_A \tag{3.3}$$

We call U *a neighbor potential with respect to a neighborhood system* $\mathcal{N}$ *if* $U_A \equiv 0$ *whenever* A *is not a clique. The function of* U_A *are then called clique potentials. In the following we denote clique potential by* U_C *where* $C \in \mathcal{C}$.

Proposition 3.1. *Let* Π *be a Gibbs field where the energy function is the energy of some neighbor potential* U *with respect to a neighborhood system* $\mathcal{N}$*, that is*

$$\Pi(x) = Z^{-1} exp\Big(-\sum_{c \in \mathcal{C}} U_c(x)\Big)$$

where

$$Z = \sum_{y \in \chi} exp\Big(-\sum_{c \in \mathcal{C}} U_c(y)\Big)$$

Then the local characteristics for any subset $A \subset S$ *are given by*

$$\Pi(X_A = x_A | X_{S\setminus A} = x_{S\setminus A}) = \frac{exp\Big(-\sum\limits_{c \in \mathcal{C}, \mathrm{c} \cap \mathrm{A} \neq \emptyset} U_c(x)\Big)}{\sum\limits_{y \in \chi} exp\Big(-\sum\limits_{c \in \mathcal{C}, \mathrm{c} \cap \mathrm{A} \neq \emptyset} U_c(y_A x_{S\setminus A})\Big)}$$

Proof: (Winkler [87] p.65)
Let $x_A = X_A(x)$ and $x_{S\setminus A} = X_{S\setminus A}(x)$.
Use a formula like $\exp(a-b)/\sum_d \exp(d-b) = \exp(a)/\sum_d \exp(d)$ to compute

$$\begin{aligned}\Pi(X_A = x_a | X_{S\setminus A} = x_{S\setminus A}) &= \frac{\Pi(X = x_A x_{S\setminus A})}{\Pi(X_{S\setminus A} = x_{S\setminus A})} \\ &= \frac{Z}{Z}\frac{\exp\Big(-\sum_{C\cap A\neq\emptyset} U_C(x_A x_{S\setminus A}) - \sum_{C\cap A=\emptyset} U_C(x_A x_{S\setminus A})\Big)}{\sum_{y_a\in\chi} exp\Big(-\sum_{C\cap A\neq\emptyset} U_C(y_A x_{S\setminus A}) - \sum_{C\cap A=\emptyset} U_C(y_A x_{S\setminus A})\Big)} \\ &= \frac{\exp\Big(-\sum_{C\cap A\neq 0} U_C(x_A x_{S\setminus A})\Big)}{\sum_{y_A\in\chi}\exp\Big(-\sum_{C\cap A\neq 0} U_C(y_A x_{S\setminus A})\Big)}\end{aligned}$$

■

3.3 Texture Modeling

A texture can be modeled [1] as a parametric family of spatially homogeneous random fields which depend on a number of hyperparameters. The choice of an appropriate texture model has two aspects: to find an appropriate model class and to identify suitable parameters,e.g.,*supervised learning*. The *unsupervised learning* can be found in, e.g., Paget and Longstaff [69]. As it is mentioned above,the random field models are more appropriate for irregular textures.

Define the potential function $U_c(x)$ as a product of a function $\tilde{U}_c(x)$ times a control parameter θ_c which depends on the type of the considered clique, an example is shown in figure. 3.4.

Then, the energy function has the form

$$H(x) = \sum_{c\in C} \theta_c \tilde{U}_c(x) \tag{3.4}$$

[1] for a detailed explanation look at Winkler, [87]

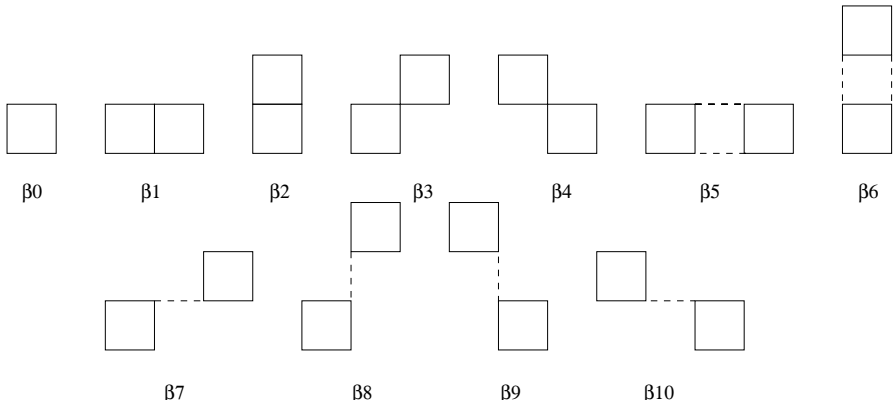


Figure 3.4: The different clique types with their associated clique parameters

For notational convenience we will denote $\tilde{U}_c$ again by U_c.

3.3.1 Random Field Texture Models

If $s, t \in S$ form a clique of type i we denote this clique by $< s, t >_i$. The expression $\sum_{<s,t>_i}$ denotes the sum over all pair cliques of type i. The number of cliques of different shape depends on the order of the neighborhood and is denoted by $\mathcal{T}(\mathcal{N}^o)$.

3.3.1.1 The Ising Model

The energy function of the Ising model is

$$H(x) = \sum_{i=1}^{\mathcal{T}(\mathcal{N}^o)} \theta_i \sum_{<s,t>_i} U(x_s, x_t) \tag{3.5}$$

where the clique potential for every pair clique C of any shape is defined by

$$U(x_s, x_t) = \begin{cases} +1, & \text{if } x_s = x_t; \\ -1, & \text{otherwise} \end{cases} \tag{3.6}$$

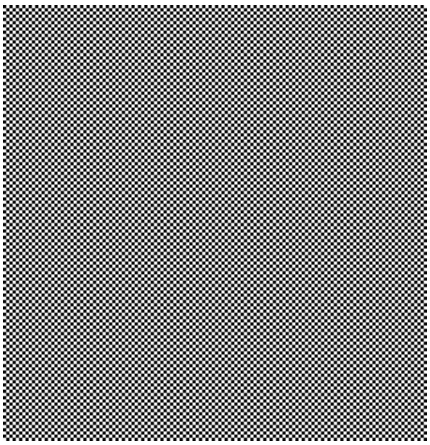


Figure 3.5: Ising Model, with number of labels = 2,$\theta_1 = \theta_2 = 1.0$ and $\theta_3 = \theta_4 = -1.0$

The local characteristics are given by

$$\Pi(x_s|x_{\mathcal{N}(s)}) = \frac{exp\Big(-\sum_{i=1}^{\mathcal{T}(\mathcal{N}^o)} \theta_i \sum_{<s,t>_i} U(x_s, x_t)\Big)}{\sum_{y_s \in X} exp\Big(-\sum_{i=1}^{\mathcal{T}(\mathcal{N}^o)} \theta_i \sum_{<s,t>_i} U(y_s, x_t)\Big)} \tag{3.7}$$

The sign and value of the θ_i in this model gives us the characteristic of this model. We can say that for $\theta > 0$ the two similar gray values in a clique of the same type will contribute a positive amount to the energy of the whole image, as a consequence the image energy will be increasing. Oppositely, two dissimilar gray values within one clique lower the total energy. If in image analysis low energy configurations are more likely than higher ones, then naturally we expect that a sample from the corresponding Gibbs distribution tends to have dissimilar gray value within the clique.

The same logic will follow for $\theta < 0$, where we expect similar gray value within the clique. The strength of the coupling depends on the magnitude of $|\theta|$, Scholz [74].

Furthermore, the Ising model has properties that the potential for a clique with two different labels is independent of the absolute difference of the labels. This means that the Ising model could not make differences of more than 2 gray values. The other is, if we consider different samples from the Ising model, especially with respect to the proportions of the gray values, it will give the same probability.

3.3.1.2 The Auto Binomial Model

The clique potentials of the pair cliques are the product of the labels at neighboring sites,i.e.

$$U_c(x) := x_s x_t \quad \text{if } c =< x_s, x_t >_i \text{ for some } i = 1, ..., \mathcal{T}(N^o)$$

The energy function is

$$H(x) = -\sum_s ln\binom{M-1}{x_s} - \theta_0 \sum_s x_s - \sum_{i=1}^{\mathcal{T}(\mathcal{N}^o)} \theta_i \sum_{<s,t>_i} x_s x_t \tag{3.8}$$

If we define $a(s) := exp(\theta_0 + \sum_i \theta_i \sum_{<s,t>_i} x_t)$ we obtain

$$\begin{aligned} \Pi(x_s|x_{\mathcal{N}(s)}) &= \frac{exp\Big(ln\binom{M-1}{x_s} - \theta_0 \sum_s x_s - \sum_{i=1}^{\mathcal{T}(\mathcal{N}^o)} \theta_i \sum_{<s,t>_i} x_s x_t\Big)}{\sum_{y_s \in X} exp\Big(ln\binom{M-1}{y_s} - \theta_0 \sum_s y_s - \sum_{i=1}^{\mathcal{T}(\mathcal{N}^o)} \theta_i \sum_{<s,t>_i} y_s x_t\Big)} \\ &= \binom{M-1}{x_s} \frac{a^{x_s}}{\sum_{y_s} \binom{M-1}{y_s} a^{y_s}} \end{aligned}$$

This model owes its name to the fact that its local characteristics reduce to binomial distributions where the probability of success is determined by the neighborhood configuration and the number of trials equal to the number of gray levels. This can be shown as, according to the binomial formula the denominator equals to $(1+a)^{M-1}$ and therefore

$$\Pi(x_s|x_{\mathcal{N}(s)}) = \binom{M-1}{x_s} \Big(\frac{a}{1+a}\Big)^{x_s} \Big(1 - \frac{a}{1+a}\Big)^{M-1-x_s}$$

is binomial with probability of success $\frac{a}{1+a}$ and number of trial M.

The parameters θ_i determine how the pixels interact locally; Acuna [2] refers to them as interaction parameters. If θ_i is positive, the pixels in the clique tend to be similarly colored, whereas negative values of θ_i increase the likelihood of a different coloring for two pixels in the same clique. Acuna also stated that this model is biased. In particular, in most instances the model tends to favor all-white over all black images.

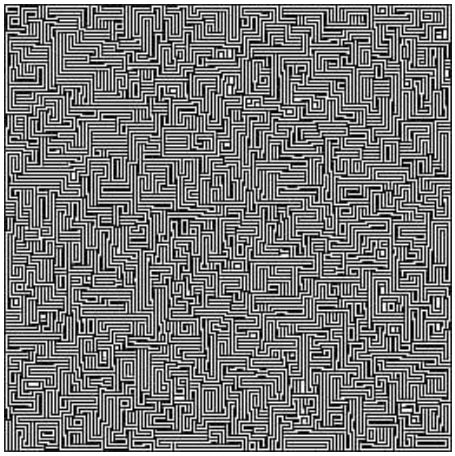


Figure 3.6: Autobinomial Model, with Number of labels = 10, $\theta_0 = -1.0$, $\theta_1 = \theta_2 = 1.0$ and $\theta_3 = \theta_4 = -1.0$

3.3.1.3 The Phi-Model

The energy function of this model is

$$H(x) = \sum_{i=1}^{\mathcal{T}(\mathcal{N}^{\circ})} \theta_i \sum_{<s,t>_i} \Phi(x_s - x_t) \tag{3.9}$$

with clique potentials of the form

$$\Phi(\Delta) = \frac{-1}{1 + |\frac{\Delta}{\delta}|^2} \tag{3.10}$$

where $\delta > 0$ is a fixed scaling parameter.

The local characteristics of the model are

$$\Pi(x_s|x_{\mathcal{N}(s)}) = \frac{exp\Big(-\sum_{i=1}^{\mathcal{T}(\mathcal{N}^{\circ})} \theta_i \sum_{<s,t>_i} \Phi(x_s - x_t)\Big)}{\sum_{y_s \in X} exp\Big(-\sum_{i=1}^{\mathcal{T}(\mathcal{N}^{\circ})} \theta_i \sum_{<s,t>_i} \Phi(y_s - x_t)\Big)} \tag{3.11}$$

The characteristic of the Phi model is the same as the two others models, that is the sign parameter θ_i will control the degree to which the neighboring pixels tend to have similar or dissimilar gray values in same manner. The advantage of this model lies in the possibility to create 'smooth' gray level if we choose the parameters appropriately.

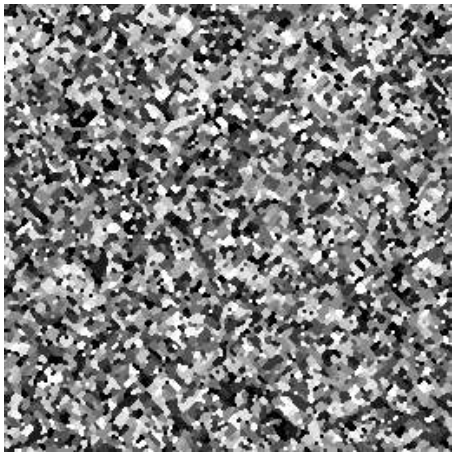


Figure 3.7: Phi Model, with number of labels=10, $\theta_1 = \theta_2 = 1.0$, $\theta_3 = \theta_4 = 0.5$ and $\theta_5 = \theta_5 = -0.5$

3.4 Synthesis for Semi Regular Texture

The models above work well for synthesizing random textures. The Phi model has been applied to tomographic image reconstruction by Geman [32], and also in segmentation of natural textured images, Geman and Graffigne [34]. The Autobinomial model which was introduced by Besag in [6], and used by Cross and Jain [18] for generating and synthesizing texture. Acuna [2] modified this model by adding a penalty term such that the Gibbs sampler can be driven towards desired proportion of intensities and keeps control of the histogram.

The Ising model seems to be simple at a first glance. But it exhibits a variety of fundamental and typical phenomena shared by many large complex systems. The simple second order neighborhood system $\mathcal{N}^2$ for 2 levels gray values and the control parameter $\theta_1 = \theta_2 = 1, \theta_3 = \theta_4 = -1$ without considering the boundary, will generate a checkerboard texture, with size of every white and black (3.5).

This can be explained as if we choose $\theta_1 = \theta_2 = 1$, then pixels with opposite gray values are favorable for the horizontal and vertical neighbors, and $\theta_3 = \theta_4 = -1$, such that diagonal neighbors tend to have *s*imilar gray values.

We modified the Ising model by changing the control parameters θ_c such that these pa-

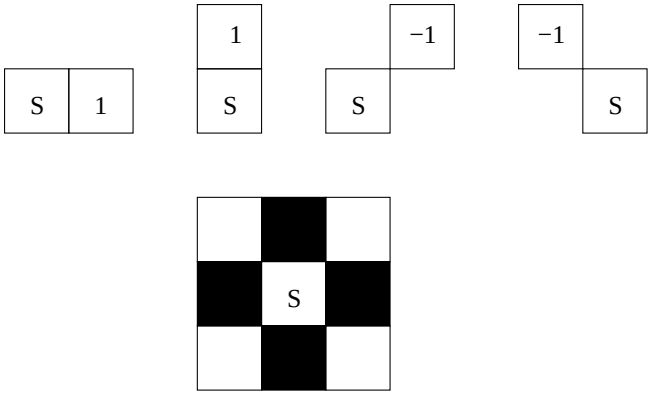


Figure 3.8: the illustration of cliques pair for $\mathcal{N}^2$ with the considering control parameters θ's to generate a checkerboard texture

rameters do not only depend on the pair of the clique types but also on the relative position of the site s in the clique. If, e.g., we consider $\mathcal{N}^2$ and the control parameter $\theta = (\theta_1, \theta_2, \theta_3, \theta_4)$ as a vector, then the position of the θ's is shown in the Figures 3.9 .

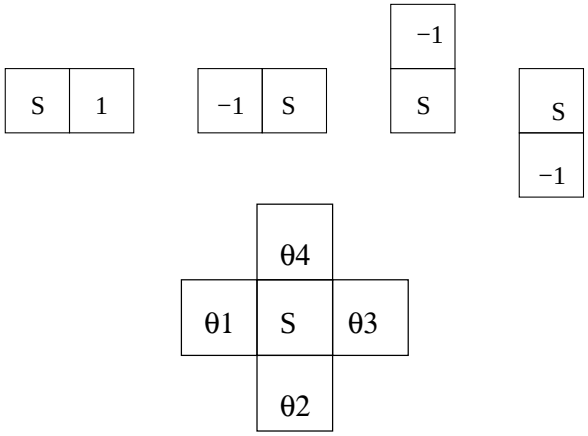


Figure 3.9: the modified cliques pair for $\mathcal{N}^1$ with the considering control parameters θ's

The generalized Ising potential function, can be written as

$$H(x) = \sum_{i=1}^{\mathcal{T}^*(\mathcal{N}^0)} \sum_{<s,t>_i} \theta_i(s,t) U(x_s, x_t) \tag{3.12}$$

where the clique potential and the control parameter for every pair clique C of any shape

is defined by

$$U(x_s, x_t) = \begin{cases} +1, & \text{if } x_s = x_t; \\ -1, & \text{otherwise} \end{cases}$$

as before in the Ising model. $\mathcal{T}^*(\mathcal{N}^0)$ is the number of the newpair clique which is equal to 2x$\mathcal{T}(\mathcal{N}^0)$ as we consider only 2-point-cliques. The control parameters $\theta_i(s,t)$ may not only depend on the type of clique but, in general, also on the gray values x_s, x_t. We give some examples below (compare, e.g. 3.14 - 3.15).

We know that the similarity or dissimilarity for a pixel to its neighborhood is measured by $U(x_s, x_t)$ and low energy configurations are more likely than the higher one. Based on these two conditions, for synthesizing binary images, $\theta_i(s,t)$ can be chosen as the negative value of the potential function $U(x_s, x_t)$, i.e.

$$\theta_i(s,t) = -U(x_s, x_t) \quad \text{or} \quad \theta_i(s,t) = -\frac{1}{U(x_s, x_t)}$$

but this implies that

$$\begin{aligned} H(x) &= \sum_{i=1}^{\mathcal{T}^*(\mathcal{N}^0)} \sum_{<s,t>_i} -1 \\ &= -\mathcal{T}^*(\mathcal{N}^0) \end{aligned}$$

and

$$\begin{aligned} \Pi(x) &= \frac{\exp(\mathcal{T}^*(\mathcal{N}^0))}{\sum_z \exp(\mathcal{T}^*(\mathcal{N}^0))} \\ &= \frac{1}{N} \end{aligned}$$

It gives the same probability for each pixel !

In Figure 3.10-Figure 3.11 we give an example for generating a binary image, i.e. a reference image, that is initiated from the binary random image. We use modified cliques pair (see Figure 3.9) and Gibbs sampler with generalized Ising potential function (3.12). First, we train the initial image to get the parameters $\theta_i(s,t)$ as the negative value of the potential function from the reference image. At first iteration, we get the right image of

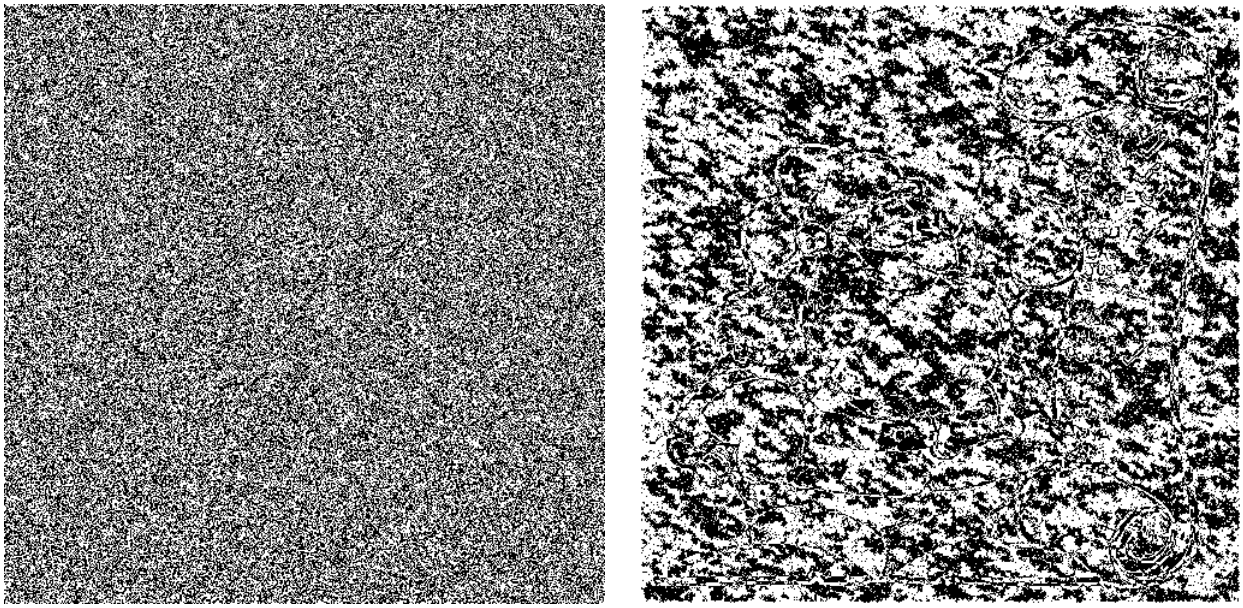


Figure 3.10: Initial image and the first iteration

Figure 3.11: The third iteration and the final iteration

Figure3.10. After several iterations, the image gets better and better until it reaches the goal, i.e, the reference image, as it is depicted in Figure 3.11.

As we stated above that the Ising model is good only for two gray levels, the Potts model gives a modification of this model, and its energy has the form

$$H(x) = -\delta \sum_{<s,t>_i} U(x_s, x_t)$$

$$U(x_s, x_t) = \begin{cases} +1, & \text{if } x_s = x_t; \\ 0, & \text{otherwise} \end{cases}$$

However, this modification is not enough to capture the differences in the gray levels, since it simply calculates the total number of similar neighbor pairs in clique. Therefore we propose another generalization of the Ising model as follows

The energy function of the generalize an Ising model is

$$H(x) = \sum_{i=1}^{\mathcal{T}^*(\mathcal{N}^0)} \sum_{<s,t>_i} \theta_i(s,t) U(x_s, x_t)$$

with potential function

$$U(x_s, x_t) = 1 - \delta|x_s - x_t|, \quad \delta > 0 \tag{3.13}$$

An example of $\theta_i(s,t)$ is

$$\theta_i(s,t) = -\frac{1}{\max^*(x_s, x_t)}, i = 1, ..., \mathcal{T}^*(\mathcal{N}^0) \tag{3.14}$$

$$max^*(x_s, x_t) = \begin{cases} 1, & \text{if } x_s = x_t; \\ \max(x_s, x_t), & \text{otherwise} \end{cases} \tag{3.15}$$

However, as Acuna states in [2], if we take only the energy function into consideration, it always produces bias, since in practice we always end up with the mean gray values if the admissible number of labels is odd and the two labels which are closest to the mean gray values, if the admissible number of labels is even. To overcome this drawback, she proposed a new energy function defined by

$$H(\mathbf{x}; \theta, \sigma, \mu) = H(\mathbf{x}; \theta) + \sigma N \|\mathbf{M}(\mathbf{x}) - \mu\|^2 \tag{3.16}$$

where N is the size of original image, $H(\mathbf{x};\theta)$ is the old energy function, $\sigma > 0$, $\mathbf{M}(\mathbf{x}) = (M_0(\mathbf{x}), ..., M_{L-1}(\mathbf{x}))$, $\mu = (\mu_0, \mu_1, ..., \mu_{L-1}) \in \mathbb{R}^L$, $\mu_l \geq 0, \sum_{l=0}^{L-1} \mu_l = 1$, and $M_l(\mathbf{x}) = 1/N^2 \sum_{s\in S} I_l(x_s)$ with $I_l(.)$ the indicator of the set$\{1\}$, and $\|.\|$ denotes the usual Euclidean norm.

The other way to control the biases of the models is through the exchange algorithm(B.3). This algorithm will keep the gray value distribution of the initial image constant throughout the iteration process, however, once the initial configuration is selected, it is not possible to move to states that have a different distribution of gray levels; i.e., the histogram of the gray lavel image is fixed.

The proposed energy function by Acuna gives a good result in the random sense if we modify the penalty term in (3.16) such that it behaves locally. In this sense, we calculate the proportion of gray levels not once through the whole sample of the image, but for every neighborhood of those pixels, i.e.,

$$H(\mathbf{x};\theta,\sigma,\mu) = H(\mathbf{x};\theta) + \sigma N_{<s,t>} \|\mathbf{M}(\mathbf{x}_{<s,t>}) - \mu_{<s,t>}\|^2 \tag{3.17}$$

where, $N_{<s,t>}$ will be the total number of cliques in the pixel's neighborhood including the pixel itself, i.e., $N_{<s,t>} = 2\mathcal{T}^*(\mathcal{N}^0) + 1$

Before simulating this model, we are going to use a gray range normalization as introduced in Gimel'sfarb [39]. Usually, the gray range $[L_{min}(x), L_{max}(x)]$ of a gray scale image x may arbitrarily change without affecting the visual perception of a texture. As regards the probability of each image sample, it should be invariant to the admissible scale changes of its gray range. The normalization transforms the initial range into a reference one $[0, Q]$ specified by a given number Q of gray values as follows

$$x_i^* = \frac{x_i - L_{min}(x)}{L_{max}(x) - L_{min}(x)} Q \quad \forall i \in \mathbb{R} \tag{3.18}$$

Remark 3.1. *This simulation depends strongly on the choice of δ, σ and the number of gray levels. So far we normalized the gray levels up to sixteen to produce some synthesizes as illustrate in Figures(3.12)-(3.14)*

Algorithm 3.1. .

1: *Normalized the sample image, with size nrow* x *mrow, using (3.18), such that its gray levels reduce up to the desired one, e.g.,* $Q = 16$. *We assume that this sample image is the smallest image which represents the whole semi-regular texture. To find this characteristic, i.e., the periodicity, in the semi regular texture, we can use the periodicity detection in Chapter 4. Based on this assumption, we can use the property of toroidal boundary to generate a continous image synthesis.*

2: *Calculate the gray level proportion of the normalized sample image.*

3: *Generate random image x with size* $M = k$ x $nrow$, $N = l$ x $ncol$, $k, l \in \mathbb{Z}$ *based on the gray level proportion which was calculated in Step 2.*

4: *Set TempConstant = L; InversTemp = log(2) / TempConstant;*
Set StopCriterion = 0.005 x M x N; *sweep = 1;*

5: *Use Metropolis sampler and Simulated annealing, Geman and Geman [33], to synthesize the texture.*
repeat

set p = 1;

for i = MBound to (M - MBound) do

for j = NBound to (N - NBound) do

5.1: *Calculate the local gray level proportion on the neigborhood of pixel (i,j);*

5.2: *Calculate the energy function* H_{old} *based on (3.17)*

5.3: *Choose a new label uniformly random.*

5.4: *Calculate the energy function for the new composition* H_{new} *based on (3.17).*

5.5: *if* $(H_{new} > H_{old})$
$p = \exp$*(-InversTemp* x $(H_{new} - H_{old}))$
we replace the old label by the new one with probability p.

5.6: *Recalculate the gray value proportion.*

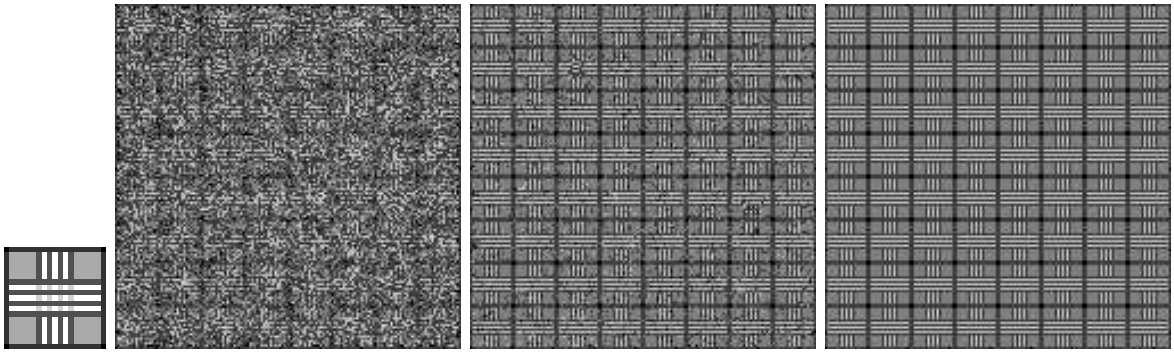


Figure 3.12: The original image in 4 gray levels, some steps of its iteration and the synthesis

5.7: *set Sweep = Sweep + 1; InversTemp =log(1+Sweep)/TempConstant.*

until (StopCriterion is fulfilled) or (x satisfies stability criterion)

The summary of stability criterion can be read, e.g. in Scholz [74].

The model in (3.12) can be used to synthesize a texture which has more than two levels of gray values by decomposing each gray level into a binary number. If we know that the maximum of the gray levels is 255 in this case we can decompose the image $\mathbf{x}$ as follows :

$$x_i = a_7 2^7 + a_6 2^6 + a_5 2^5 + a_4 2^4 + a_3 2^3 + a_2 2^2 + a_1 2^1 + a_0 2^0 \quad i = 1, ..., n \tag{3.19}$$

with $a_0, ..., a_7 \in \{0, 1\}$. So for every single pixel from the original image, we get eight compositions of the binary number derived from its coefficient, $a_0, ..., a_7$. Collect every coefficient separately, it will give a binary image and at the end we have eight binary images. Since now the gray level has become two then we can use (3.12) to synthesis those images one by one and we repeat this pattern to produce a larger image.

Finishing with binary images synthesizing, again we use (3.19), to join this images into the gray level from where they were decomposed.

Remarks 4

1. This method just simply produces the repetition of the original image, therefore we do not recommend to use it.

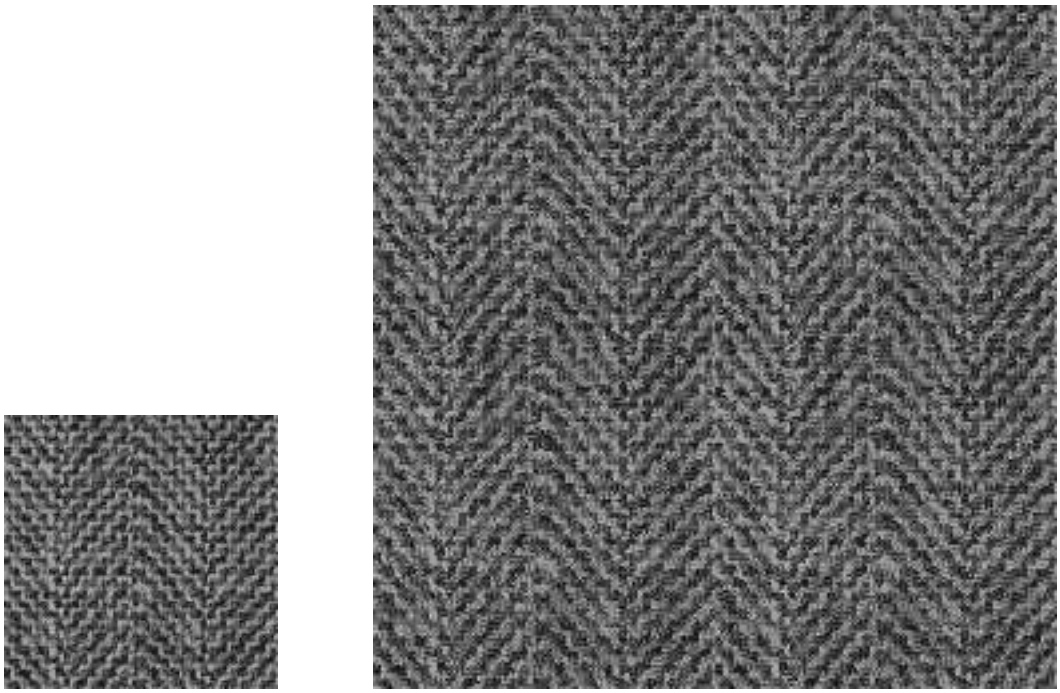


Figure 3.13: Herringbone in 16 gray levels and its synthesis

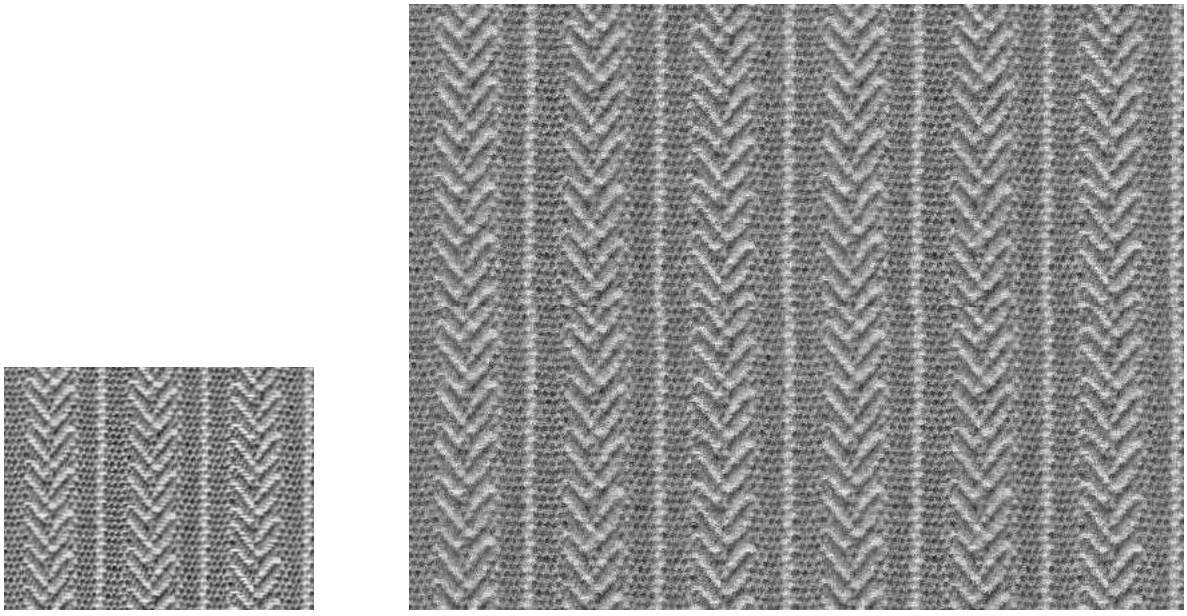


Figure 3.14: A textile in 16 gray levels and its synthesis

2. The last two binary images just perform a very random image, therefore it could be better to simulate this part separately,i.e., the last 2 images we simulate by normal Markov Random Field and just add it to the last parts of the join binary image.

Remark 5

It can be seen that sometimes the synthesizing textures have a drawback in the boundary. Look at, for example, the Herringbone in Figure.3.17. This drawback can be reduced using some technique, that will be explained in Chapter 4.

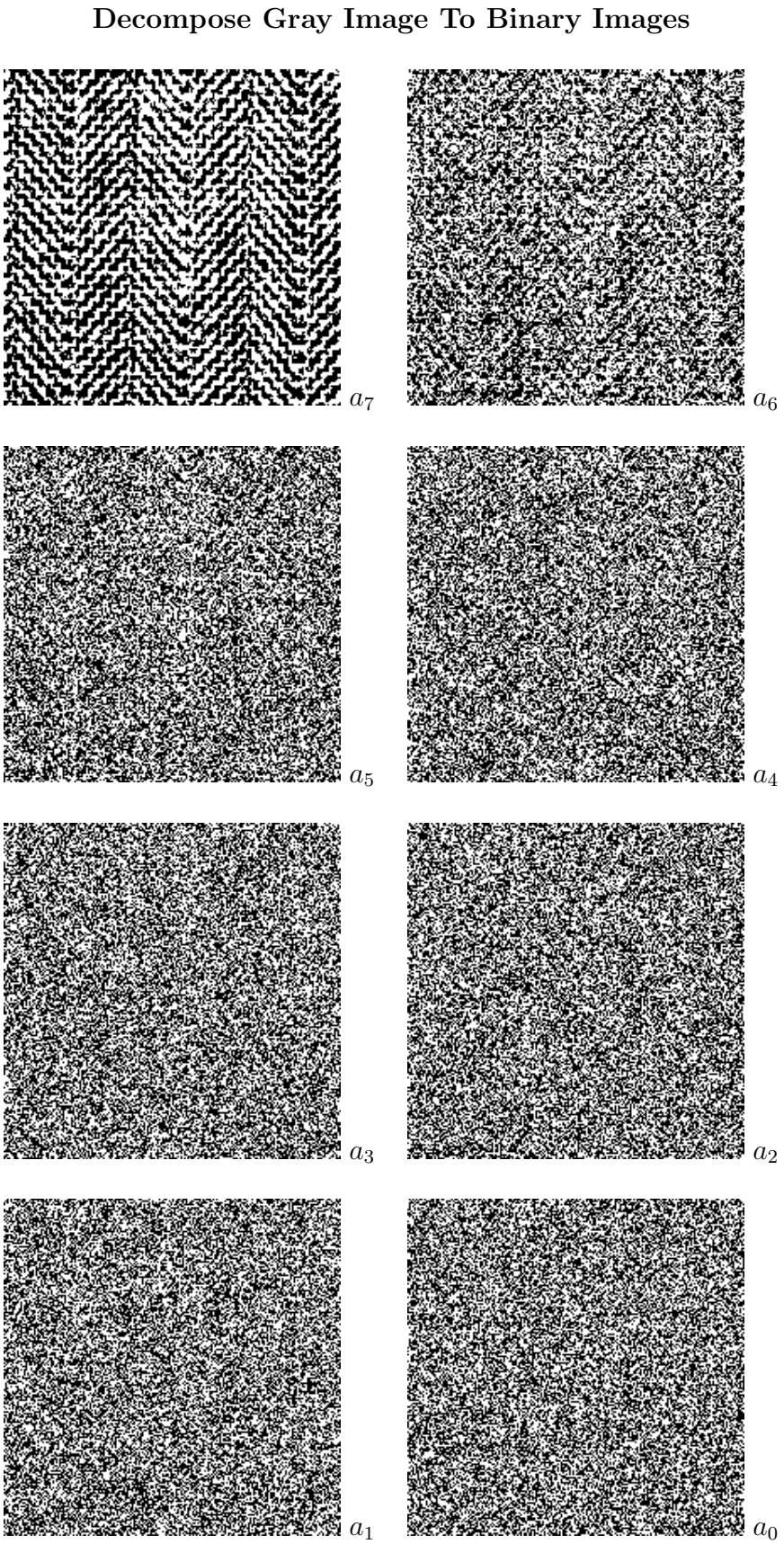


Figure 3.15: The decomposition of an image to its binary images based on (3.19)

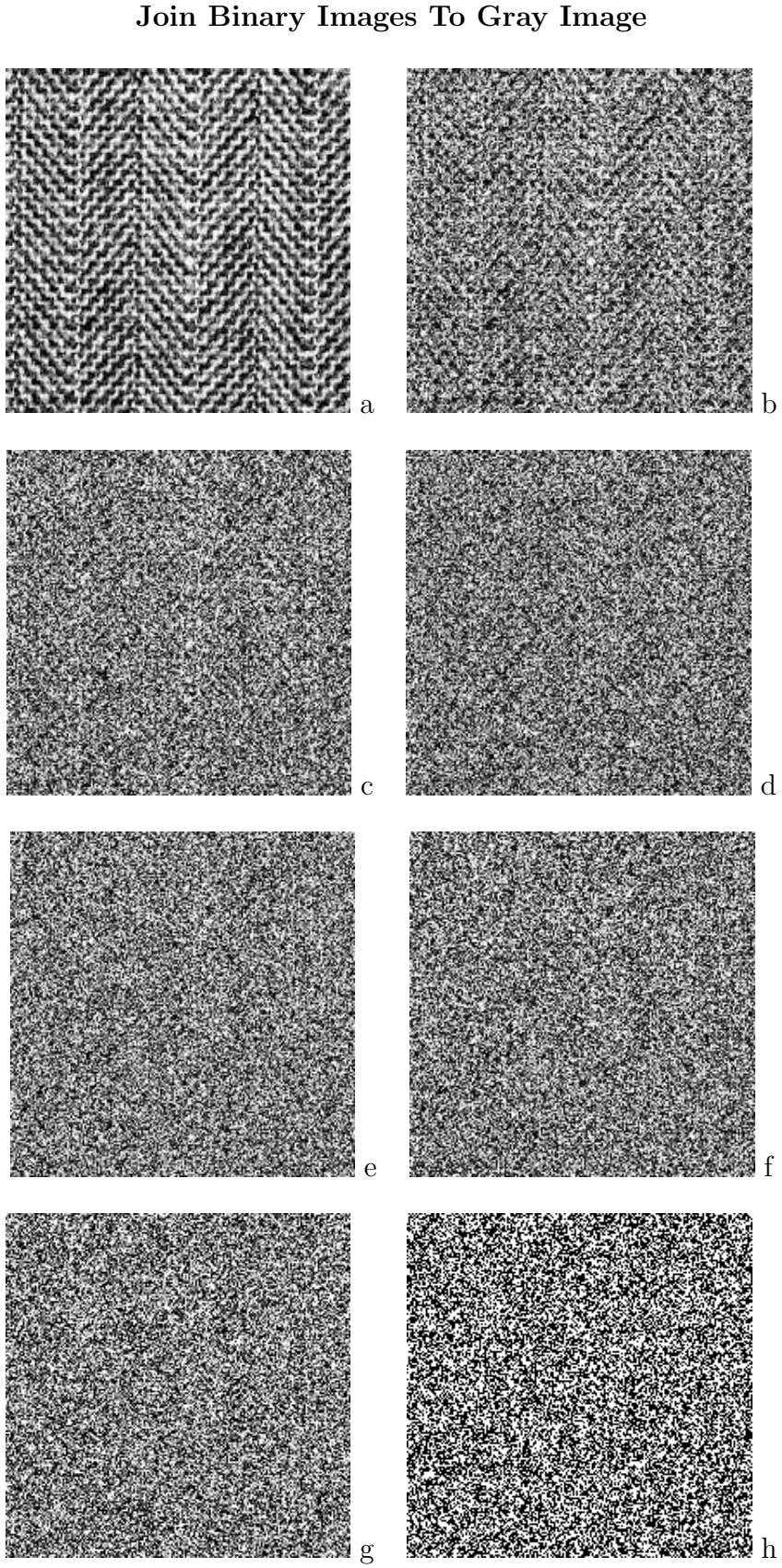


Figure 3.16: Image h is from the last decomposition, i.e. $a_0 2^0$, g is the join of the last two images,i.e., $a_0 2^0 + a_1 2^1$, and so on, the Image a will be the join from the all levels, i.e. $a_0 2^0 + a_1 2^1 + ... + a_7 2^7$

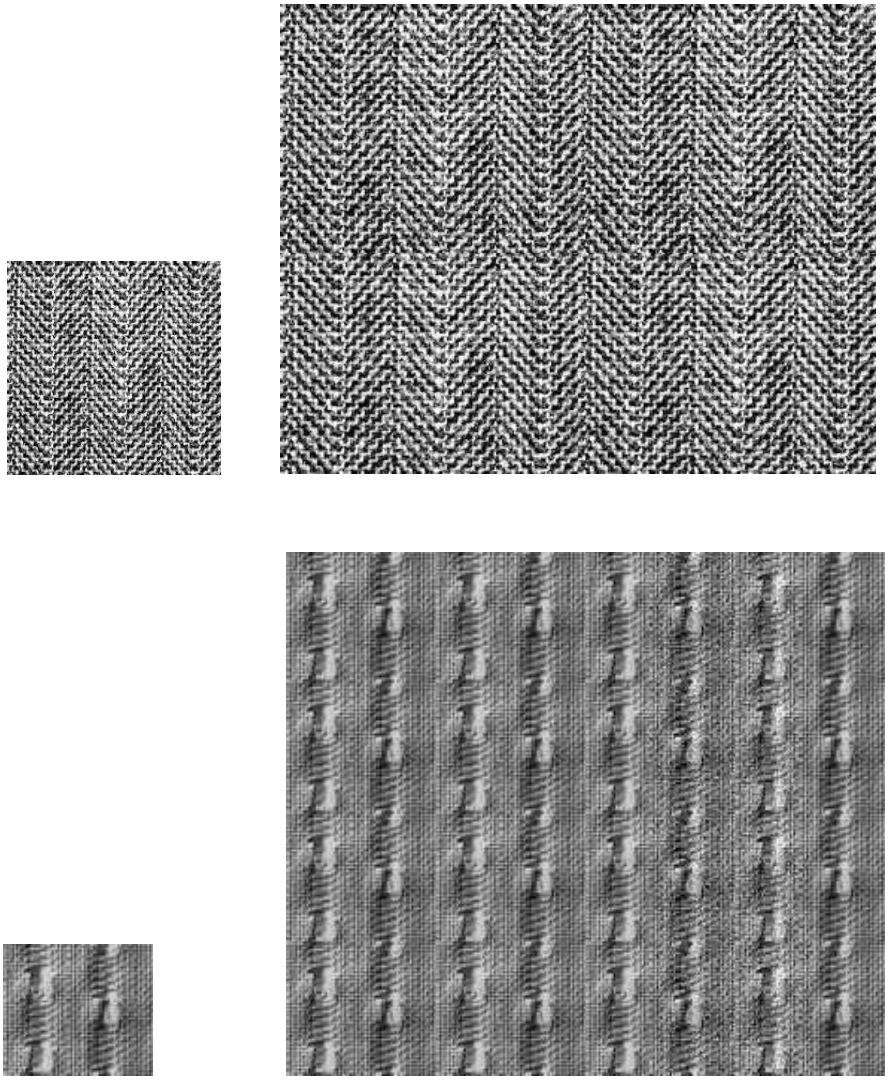


Figure 3.17: Examples of synthesizing texture using decompose and join binary images

Chapter 4

Bootstrap Simulation of Semi Regular Textures

A basic feature of many textures is spatial self-similarity, that is, similarity of certain pixel neighborhoods acting as local prototypes. A texture is formed by their spatial replication under specific deterministic or stochastic rules of mutual placement, Gimel'farb [37].

To capture this similarity, Gimel'farb introduced a so called model-based interaction map (MBIM). The MBIM indicates the partial interaction energy for a particular intra-clique shift along the choosing window. Using its maximum energy and the inter-pixel shift for the clique family yielding this maximum, two clusters or blocks pattern were selected and a texture is simulated by mutual placement of those Gimel'farb [37]. Some further developments of this technique by Gimel'farb, et.al, can be found, for examples, in ([92], [38]). Another method to synthesize a texture using a block pattern was developed by Efros and Freeman, the so called image quilting [24]. In this method, the blocks and their sizes are taken randomly from the input image, then patched together with specific measurements to get the simulated texture.

In this work we are going to use a statistical tool to simulate textures with spatial self-similarity. We consider an image as a two dimensional stochastic process and use spatial periodogram to detect the periodicity or the regularity of the image. This periodicity in two dimensions will indicate the smallest size of an image block that has self-similarity.

Using this periodicity, we are going to divide the image into blocks and regard them as sample blocks. Based on these sample blocks we are going to generate an image synthesis using bootstrap techniques with specific measurement to glue each sample data to obtain a similar texture.

We know that the bootstrap method which was introduced by Efron [23] works well for independent data. However, for bootstrapping a structure which has dependent characteristics, e.g. a stationary time series or a periodic dependent structure such as a semi regular texture, the bootstrap method has to be modified. For this purpose we only consider the bootstrap for a periodic dependent structure. Politis in [70] derived a technique to resample time series with seasonal components. This technique works for one dimensional data and we are going to derive a similar new technique to resampling the two dimensional stochastic process with periodic components to synthesize a semi regular texture.

4.1 One Dimensional case

We consider observations of the form $X_1, ..., X_N$ as random variables generated from a periodic function plus stationary noise, i.e.

$$X_t = f(t) + \epsilon_t \tag{4.1}$$

where $f(t+T) = f(t)$, $t = 1, ..., N$, is a periodic function with periodicity T, and ϵ_t is a stationary process.

4.1.1 Hidden Periodicity Detection

As a natural part of this problem the periodicity detection as the crucial step has to be considered. To obtain the periodicity, following Chiu [15], we approximate the (4.1) by a harmonic regression as follows:

$$\begin{aligned} X_t &= \mu + S(t) + \epsilon(t), \quad t = 1, ..., N; \qquad (4.2) \\ \epsilon(t) &\sim \mathcal{N}(0, \sigma^2) \\ S(t) &= \sum_{k=1}^{K} (A_k \cos \omega_k t + \phi_k) \qquad (4.3) \end{aligned}$$

consist of K sinusoidal waves at frequencies $\omega_k \neq 0$ with amplitudes $A_k \neq 0$, phases ϕ_k. K can be equal to zero, to include the case of no periodic component. Without loss of generality we assume $\mu = 0$. The periodogram is used to search for hidden periodicities and naturally it became a tool for testing the presence of periodic components.

Definition 4.1. *The periodogram of a series X_t is defined as*

$$I_x(\omega) = d_x^{(N)}(\omega) d_x^{(N)}(-\omega)/2\pi N \qquad (4.4)$$

where

$$d_x^{(N)}(\omega) = \sum_{t=0}^{N-1} X_t exp(-i\omega t)$$

is the finite Fourier transform of the series.

Let $I_x(\omega_j)$ denote the periodogram of X_t at the Fourier frequencies

$$\omega_j = (\frac{2\pi j}{N}), j = 0, 1, ..., [N] \qquad (4.5)$$

where $[N] = N/2 + 1$ if N is even and $[N] = (N + 1)/2$ if N is odd. Ordering the $I_x(\omega_j)$ ascending, we get $I_1 < I_2 < ... < I_{[N]}$.
Then the original Fisher $F - statistic$ can be defined as :

$$F = \frac{I_{[n]}}{\sum_{j=1}^{[N]} I_x(\omega_j)} \qquad (4.6)$$

This statistic is not robust since we can see that Fisher's test statistic is proportional to the maximum of the periodogram ordinates normalized by the sample mean of the periodogram ordinates, and we know that the sample mean is sensitive to outliers. Chiu [15] suggests two new statistics to test the null of

$$H_0 : \text{zero harmonics in } X_t$$

against the alternative

$$H_1 : h \text{ harmonics in } X_t \text{ , } (h \geq 1 \text{ fixed})$$

The statistics are defined as follows:

$$U(h) = \frac{I_{[N]-h+1}}{\sum_{i=1}^{[N]} I_i} \tag{4.7}$$

$$V(h) = \frac{I_{[N]-h+1}}{\sum_{i=1}^{[N]-h} I_i} \tag{4.8}$$

The (4.7) was considered in Shimshoni [78] and Lewis and Fieller [60] for detecting k peaks or outliers. However, from the earlier argument,i.e., the sample mean is not robust, Chiu excludes the r biggest periodogram ordinates from the denominator from (4.7) and gets (4.8) instead. The asymptotic distribution of $U(.)$ and $V(.)$ for testing H_0 are derived as follows:
Define

$$Z_1(h) = [N]U(h) - ln([N] - h + 1) \tag{4.9}$$

$$Z_2(h) = c([N] - k)V(k) - ln([N] - h + 1) \tag{4.10}$$

$$\text{where } \; c = 1 + \frac{h(ln(\frac{h}{[N]}))}{[N] - h} \tag{4.11}$$

Also let:

$$P_i(h) = exp\{-exp(-Z_i(h))\} \sum_{j=0}^{h-1} exp\{-jZ_i(h)/j!\}, \quad i = 1, 2 \tag{4.12}$$

If the selected level of significance is α, we reject H_0 in favor of H_1 based on the test statistic $P_i(h), \quad i = 1, 2$, if

$$P_i(h) > 1 - \alpha \tag{4.13}$$

By these statistics we detect the periodicity, and we can derive the period as follows. Let $\omega^* = \arg_\omega \max I_x(\omega_i)$ where $i = 1, \ldots, [N]$, where ω_i is the angular frequency. Then

$\hat{T} = \frac{2\pi}{\omega^*}$ is an estimate of T

In this case we take h = 1, just to test the existence of periodicity in the image.

Remark 4.1. *In this case we take h = 1, just to test the existence of periodicity in the image. However, the number of periods in the image, of course, depends strongly on the sample size of the image; it can be more than hundreds. The factorial problem in Chiu's test statistic for large factorials can be handled by simply working with logs, e.g.*

$$50! = exp(log(50!)) = exp(log1 + log2 + ... + log50)$$

4.1.2 Bootstrap for Seasonal Time Series

We partition the sample into blocks of length T, such that we get $n = \lfloor N/T \rfloor$ original blocks, where $\lfloor . \rfloor$ denotes the integer part.

The seasonal block bootstrap algorithm is defined below

Definition 4.2. *: Bootstrap for Seasonal Time Series(Politis [70])*

- *Let k be the number of desired bootstrap blocks*
- *Draw $b_0, ..., b_{k-1}$ i.i.d uniformly distributed on the set $\{0, .., n-1\}$*
- *The bootstrap data $X_1^*, X_2^*, ..., X_{kn}^*$, are generated from the original sample, such that*

$$X^*_{mT+j} = X_{b_m T+j} \tag{4.14}$$

where $m = \{0, \ldots, k-1\}$ and $j = \{1, \ldots, nk\}$

Remark 4.2. *This works only for fixed period and a good signal to noise ratio.*

Example 1. *As an illustration we give a simple simulation study. We generate a random series from a Sinus function with $t = 1, ..., 310$ and periodicity $T = 62$. We perturb this series by adding correlated errors, which are generated as a moving average from a gaussian distributed random variable with mean zero and variance = 2, as follows:*

$$X_t = \sin(\frac{2\pi t}{T}) + \epsilon_t$$

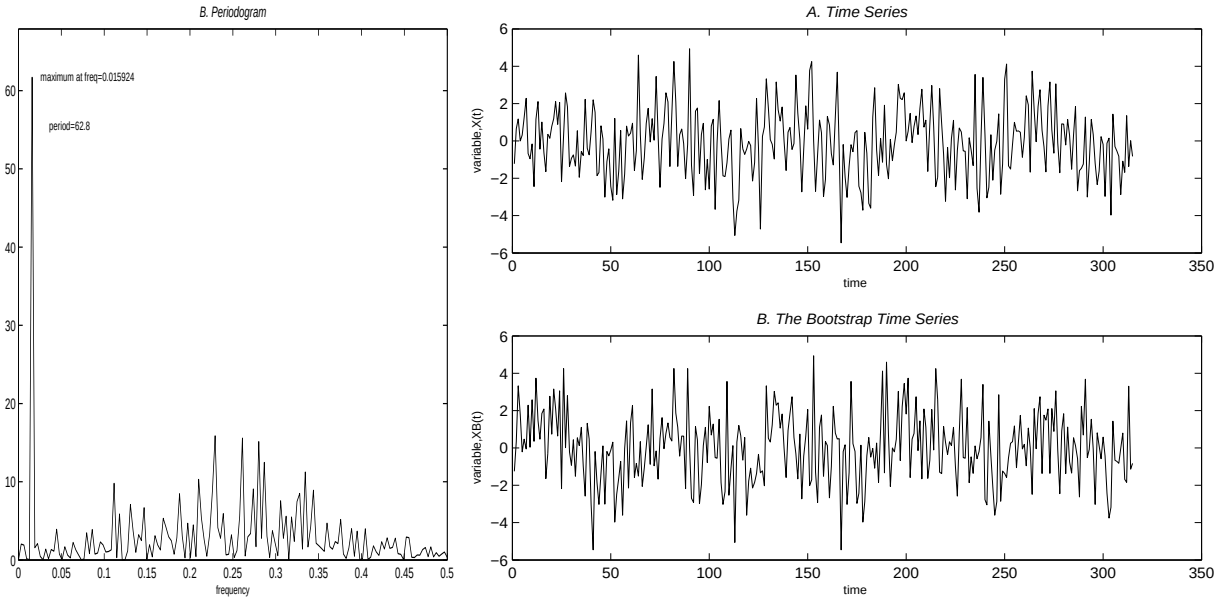


Figure 4.1: The Period,the original sample, and the bootstrap sample for 1D case

where, the ϵ_t *are generated as follows:*

$$\eta \sim \mathcal{N}(0, 2);$$
$$\epsilon_t = 0.25\eta_t - 0.5\eta_{t-1};$$

The periodogram of this series is maximum at frequency 0.015924, it implies that the periodicity is 62.8 (i.e. 1/0.015924). We use the round version of this periodicity to simulate a series using one dimensional bootstrap procedure for the same length. Figure 4.1 shows the periodogram, the original series and the simulated one. We can see that the simulated version can imitate the original one reasonably.

4.2 Two Dimensional Case

Analogously to the one dimensional case, we consider a semi regular image as a two dimensional stochastics process which has a hidden period in it,i.e,

$$X(t_1, t_2) = f(t_1, t_2) + \varepsilon(t_1, t_2),\ t_1 = 1, \ldots, M; t_2 = 1, \ldots, N \tag{4.15}$$

where the noise is part of a stationary random field $\varepsilon(t_1, t_2),\ -\infty < t_1, t_2 < \infty$, with mean 0 and finite variance. We assume that the function f has periodicity property, i.e., $f(t_1 + T_1, t_2 + T_2) = f(t_1, t_2) \forall t_1, t_2$ and we use two dimensional periodogram to detect the hidden period, i.e, T_1, T_2 from $X(t_1, t_2)$.

Remark 4.3. *The same test as in the one dimensional case is used to detect the hidden periodicity in the two dimensional case as the two-dimensional periodogram shows the same asymptotic behaviour as the one dimensional periodogram.*

Definition 4.3.

$$I_x(\omega_1, \omega_2) = d_x^{(M,N)}(\omega_1, \omega_2) d_x^{(M,N)}(-\omega_1, -\omega_2)/2\pi MN \tag{4.16}$$

where

$$d_x^{(M,N)}(\omega_1, \omega_2) = \sum_{t_1=0}^{M-1} \sum_{t_2=0}^{N-1} X_{t_1,t_2} exp(-i\omega_1 t_1 - i\omega_2 t_2)$$

is the Fourier transform of the series.

Now we use the two dimensional periodogram to detect the hidden periodicities, and we let $(\omega_1^*, \omega_2^*) = \arg_{\omega_1,\omega_2} \max I_x(\omega_i, \omega_j),\quad i = 1, \ldots, [M],\quad j = 1, \ldots, [N]$, where $I_x(\omega_i, \omega_j)$ is the periodogram and ω_i, ω_j are the angular frequencies.

Then the periodicity estimates are

$$\hat{T_1} = \frac{2\pi}{\omega_1^*} \qquad\qquad \hat{T_2} = \frac{2\pi}{\omega_2^*}$$

Obtaining the $\hat{T_1}$ and $\hat{T_2}$, we can define a small block pattern of image with a size $\hat{T_1} \cdot \hat{T_2}$, based on this small image's block, we simulate the texture using a bootstrap procedure as follows:

Algorithm 4.1. *: 2D Weighted Bootstrap-Methods Algorithm*

1. *Let* $n_1 = \lfloor M/T_1 \rfloor$ *and* $n_2 = \lfloor N/T_2 \rfloor$.

2. M_b^* *and* N_b^* *are numbers of bootstrap blocks, rowwise and columnwise respectively.*

3. *Let the* $c_0, ..., c_{M_b^* * N_b^* - 1}$ *be drawn i.i.d with uniform distribution on the set* $\{0, .., n_1 * n_2 - 1\}$.

4. *Let* $\bar{\mu} = \mathbb{E}X(t_1, t_2)$ *and* $\mu_{i,j} = \frac{1}{4}(X(i-1, j-1) + X(i-1, j) + X(i, j-1) + X^*(i, j)$

   4.1. *repeat*

   4.2. *generate* $X^*(1,1), X^*(1,2), ..., X^*(n_1 M_r, n_2 N_r)$ *where*

   $$X^*(kT_1 + i, lT_2 + j) = X(c_{l(M_b^* - 1) + k} + i, c_{l(M_b^* - 1) + k} + j) \tag{4.17}$$

   where $k = 0, \ldots M_b^* - 1, l = 0, \ldots, N_b^* - 1$ *and*
   $i = 1, \ldots, T_1, j = 1, \ldots, T_2$

   4.3. *until* $\mu(kT_1 + i, lT_2 + j) < const \cdot \overline{\mu}$
   where $0 < const \leq 1$ *measures the similarity of the bootstrap synthesis to the original image.*

In the last step we give a restriction to the generated bootstrap samples such that their local sample mean will not excede the sample mean of the image. The aim of this bound is to *glue* the fragments of the bootstrap sample, such that the similarity between these two images can be achieved.

If $const = 0$, then we only copy the image and this bootstrap procedure will mean nothing. However, the step 4.3 is sometimes quiet hard to fulfill. It depends on the characteristic of the randomness and the regularity of the image. If step.4.3 could not be fulfilled after, let say 10 to 20 times iterations, then we give two suggestions

- we let $X^*(kT_1 + i, lT_2 + j) = X(c_{l(M_b^* - 1) + k} + i, c_{l(M_b^* - 1) + k} + j)$ regardless of the restriction, and then modelled the error based on spatial error model. This will be discussed in the next section.

- set $X^*(kT_1+i, lT_2+j) = X(T_1+i, T_2+j) + U(0, \mu(kT_1+i, lT_2+j))$. It means that we copy the original pixel in that particular position and add an uniformly random number between zero to its local sample mean.

We give an illustration of the bootstrap procedure for synthesizing semi regular texture in Figure.4.2 to Figure.4.5. We have our original texture as in Figure.4.5, the Fourier transform of its sample correlation, i.e., the periodogram is shown in Figure.4.2 left. Due to the Fourier transform property we get four quadrants. The point in the middle of Figure.4.2 is the origin. Now, in every quadrant we threshold the periodogram with the thresholding point is the maximum periodogram in the quadrant. We have now, $(\omega^*_{11}, \omega^*_{12})$, $(\omega^*_{21}, \omega^*_{22})$, $(\omega^*_{31}, \omega^*_{32})$, $(\omega^*_{41}, \omega^*_{42})$. Using the relation that $T = \frac{\omega*}{2\pi}$, we get (T_{11}, T_{12}), (T_{21}, T_{22}), (T_{31}, T_{32}), (T_{41}, T_{42}) as in Figure.4.2 right.

For a texture where the randomness is blended well with the regularity, like in this example, we can choose one of the periodicity that appears in the four quadrants arbitrarily. In Figure.4.4 the size of the blocks clockwise are 11x12, 15x11, 13x13 and 14x13 represent the periodicities that appear in Figure.4.2, we can see shifting 1 to 5 pixels in this example does not effect much to the appearance of the simulation's result.

However, the drawback of our methods for this moment is the lack of rotation invariant. We show this problem by an image of a chess board which is rotated 45 degrees. Bootrapping directly will give noisy image, e.g., gray spots in the white area. In the original image the gray pixels are located in the edges or in the corner of the blocks and the control of this location is lost in the simulation such that the gray pixels appear in the white area. Beside that, we also can see that the edges between "black and white" are no longer straight line but a little bit wiggly. This happend because of the interchanging of the white to the black pixels improperly.

So far what we can do is a semi manual approach as we show in Figure.4.7 to Figure.4.10. First, we calculate the angle from the periodogram by thresholding it until we get two principal points Figure.4.7, let say (x_1, y_1) and (x_2, y_2) so that we can calculate the angle by

$$\alpha = \arctan(\frac{y_2 - y_1}{x_2 - x_1}) \tag{4.18}$$

We rotate the image and once more compute the periodicities of the rotated image, do the block bootstrap and get the synthesis image as we did for synthesizing, e.g., Figure.4.2.

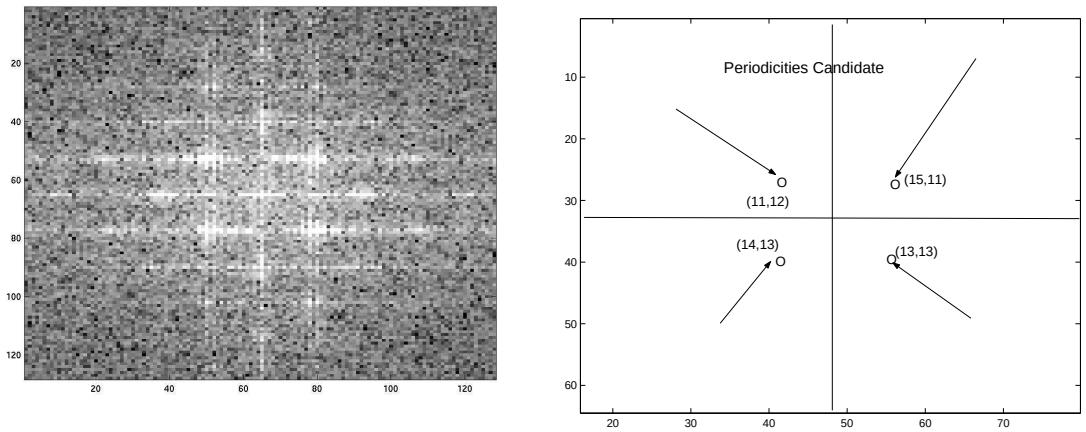


Figure 4.2: The Periodogram and The Threshold Periodogram of The Image

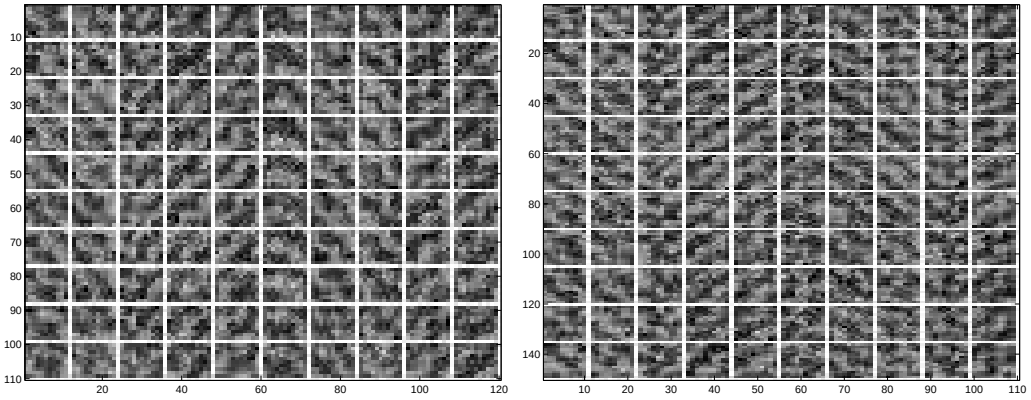


Figure 4.3: The simulated bootstrap with block of size 11x12 (left) and 15x11 (right)

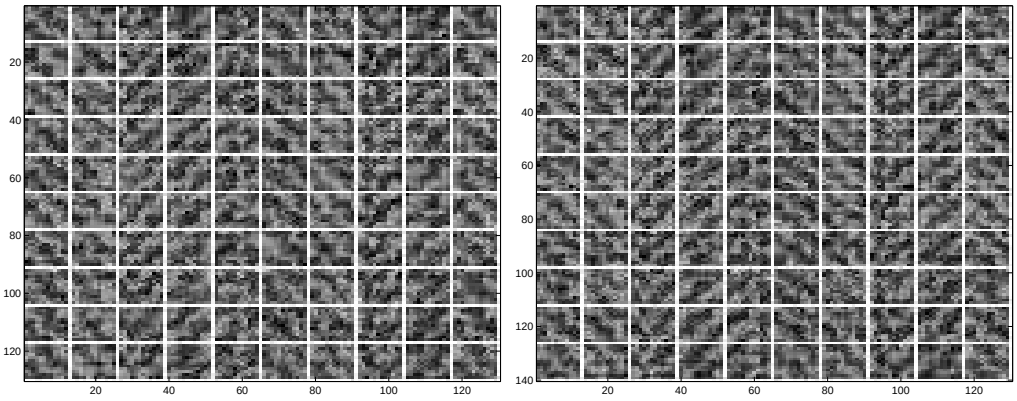


Figure 4.4: The simulated bootstrap with block of size 13x13 (left) and 14x13 (right)

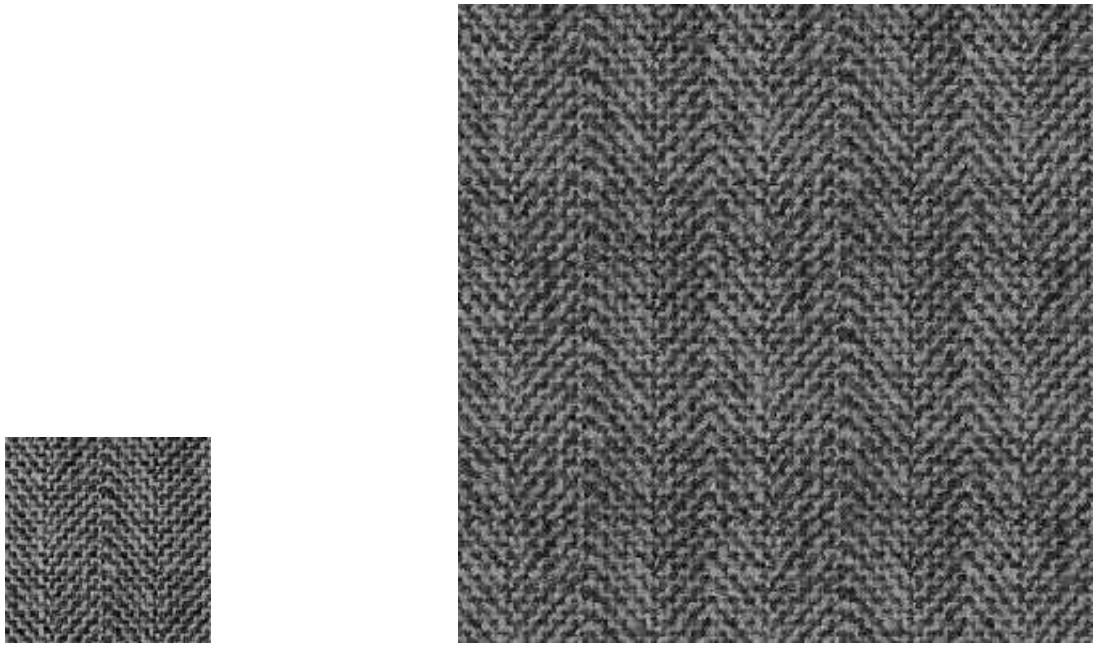


Figure 4.5: 'Herringbone' and The Synthesis one

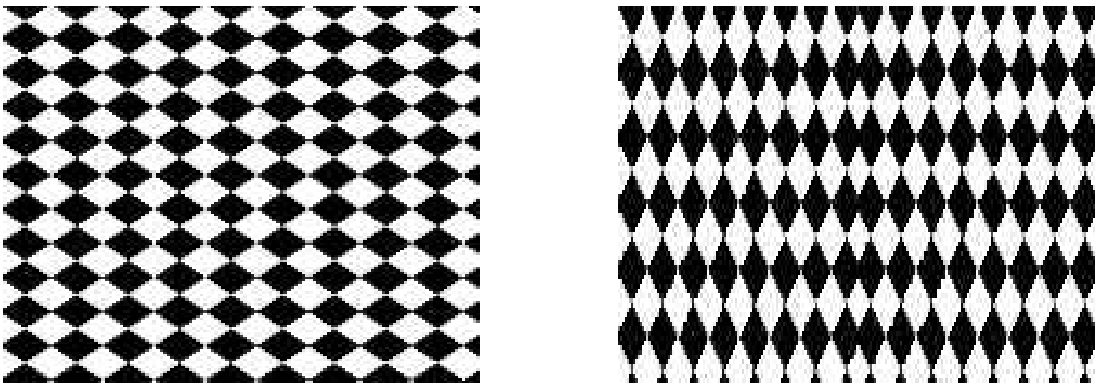


Figure 4.6: Bootstrapping rotated 45 degrees chess board

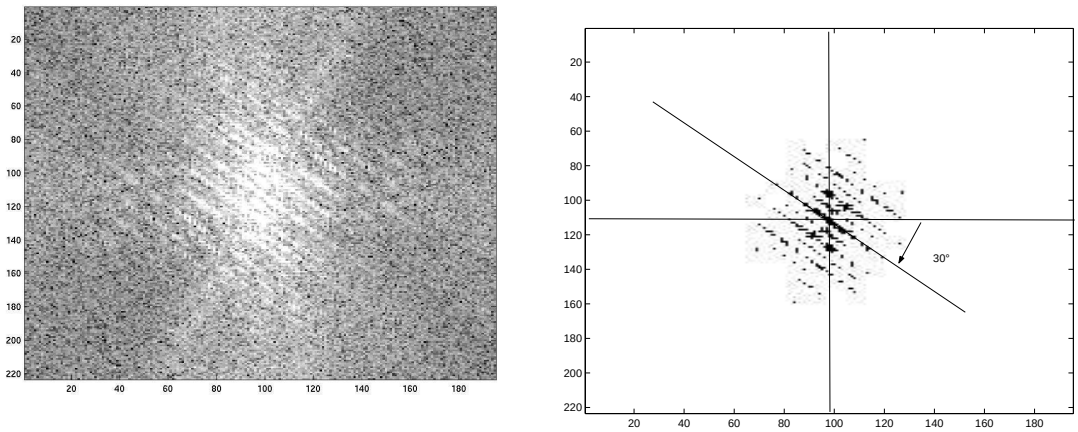


Figure 4.7: The Periodogram and The Threshold Periodogram of The Image

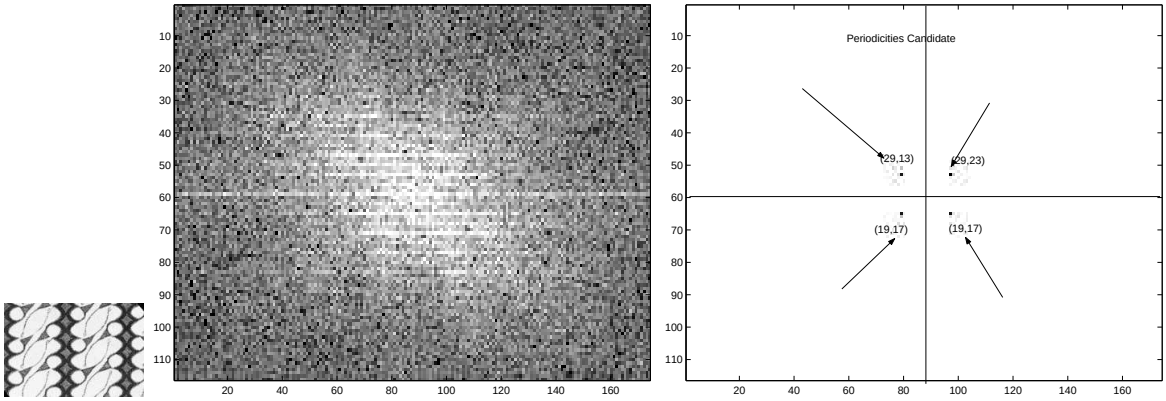


Figure 4.8: The Rotated, The Periodogram and The Threshold Periodogram of The Image

To give a clear description we illustrate in the following figures the steps that we use to simulate an image which has an angle in its feature. First, we rotate the original image with the angle that we get from (4.18), see Figure.4.8-left and then again we compute the Periodogram and its Threshold as in Figure.4.8-middle and right. blur

This time we have three choices of periodicities which give different appearance in the simulation as we can see in Figure.4.9. For this simulation we choose the 29x13 block, since for larger simulation it gives the best result from the others two, i.e., the 29x23 and the 19x17 block respectively.

Finally, we transfer back this synthesis image to the original angle to get the similar image as the original one, see Figure.4.10

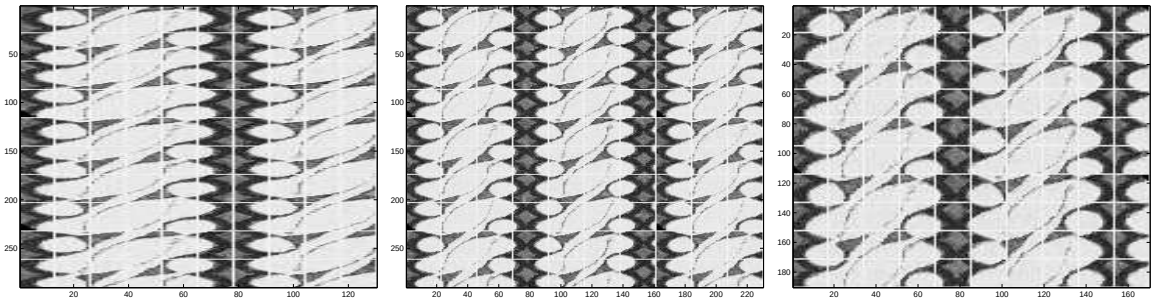


Figure 4.9: The simulated bootstrap with block of size 29x13 (left), 29x23 (middle) and 19x17 (right)

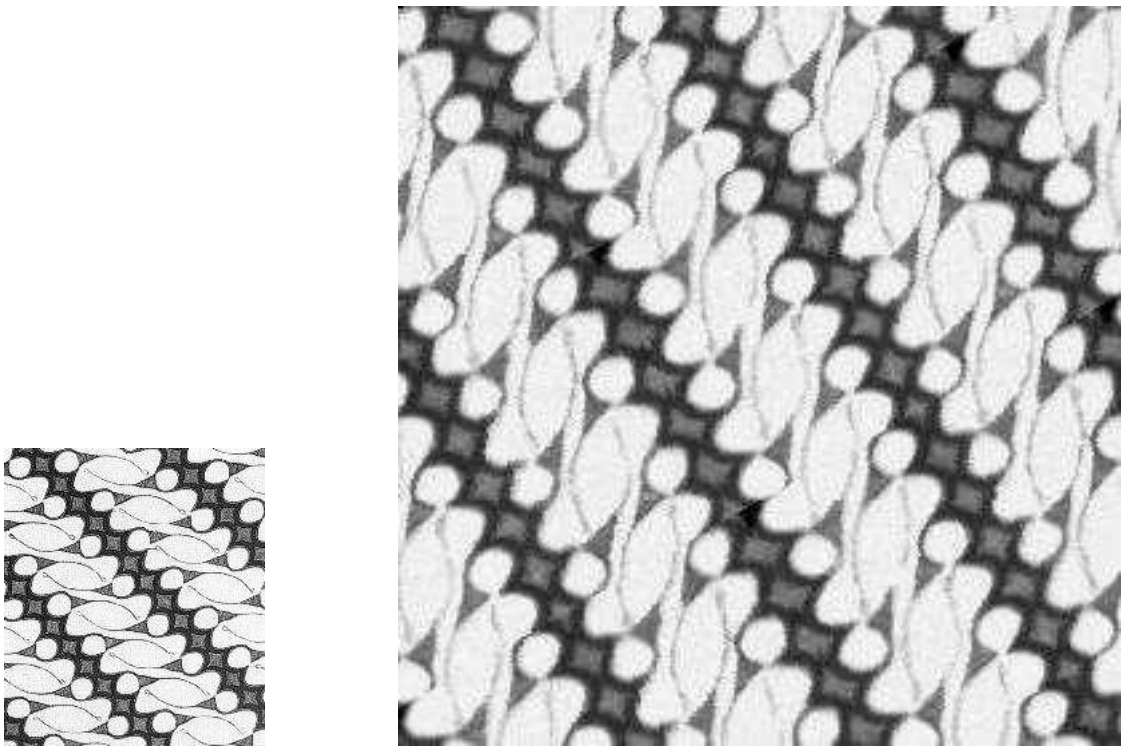


Figure 4.10: 'Parang Rusak' and The Synthesis one

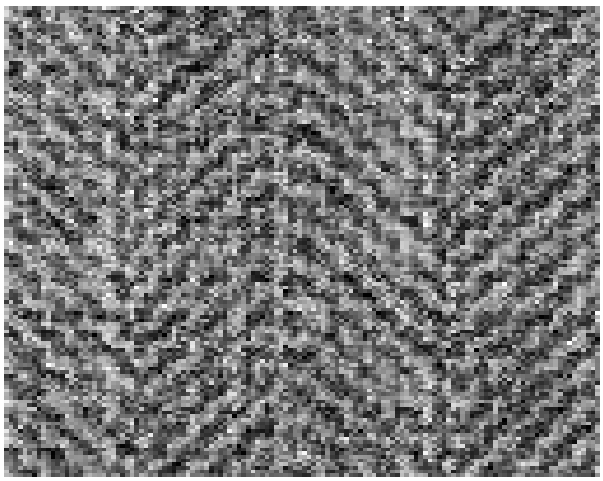


Figure 4.11: An example of blurring in the bootstrap synthesis

4.3 Bayesian Spatial Autoregressive models

In this section we are going to reduce the blurring effect which can be observed e.g. in Figure 4.11 by modelling the residual error as a Bayesian Spatial Autoregressive models, Le Sage [59]. This model is an extended version of the Spatial Autoregressive models that can be found in, e.g., Anselin [4]. However, before we model the residual error let us review some terminologies in spatial autoregressive models and Bayesian analysis.

We remark this method gives a good result if it is applied to the texture which both randomness and regularity are blended well, e.g., the Herringbone texture.

4.3.1 Spatial Autoregressive Models

We consider a spatial stochastic process typically observed on a rectangular past of the integer lattice $\mathbb{Z}^2$. We enumerate the observations to get them in vector form $y = (y_1, ..., y_n)'$

Definition 4.4. *:General First order Spatial Autoregressive models*
A class of spatial autoregressive models taking the form as follow

$$\begin{aligned} y &= \rho W_1 y + Z\beta + u \\ u &= \lambda W_2 u + \epsilon \\ \epsilon &\sim \mathcal{N}(0, \sigma^2 I_n) \end{aligned} \tag{4.19}$$

where y contains an n x 1 vector of cross-sectional dependent variables and Z represent an nxk matrix of explanatory variables. W_1 and W_2 are known n x n are spatial weight matrices, β is a k x 1 vector of parameters, ρ, λ are real-valued parameters.

Example 2. *: Some derived models*

- *Set* $Z = 0$ *and* $W_2 = 0$ *produces a first-order spatial autoregressive model*

$$\begin{aligned} y &= \rho W_1 y + \epsilon \\ \epsilon &\sim \mathcal{N}(0, \sigma^2 I_n) \end{aligned} \tag{4.20}$$

  In this model y is explained to be a linear combination of the contiguous or neighboring units with no other explanatory variables.

- *Set* $W_2 = 0$ *produces a mixed regressive-spatial autoregressive model*

$$\begin{aligned} y &= \rho W_1 y + Z\beta + \epsilon \\ \epsilon &\sim \mathcal{N}(0, \sigma^2 I_n) \end{aligned} \tag{4.21}$$

  Here we have additional explanatory variables in the matrix Z to explain variation in y over the spatial sample of observations.

- *Set* $W_1 = 0$ *results in a regression model with spatial autocorrelation in the disturbances*

$$\begin{aligned} y &= Z\beta + u \\ u &= \lambda W_2 u + \epsilon \\ \epsilon &\sim \mathcal{N}(0, \sigma^2 I_n) \end{aligned} \tag{4.22}$$

4.3.1.1 The Estimation

Lemma 4.1. *: Least Square Estimation*
Consider the First Order Autoregressive model in eq.(4.20). Applaying least squares will produce an estimate for the single parameter ρ *:*

$$\hat{\rho} = (y'W'Wy)^{-1}y'W'y \tag{4.23}$$

Proof. By direct calculation we get

$$\epsilon = y - \rho W y$$

take the minimum least square

$$\begin{aligned} \| \epsilon \|^2 &= \min_{\rho} (y - \rho W y)'(y - \rho W y) \\ \frac{\partial \epsilon^2}{\partial \rho} &= y'W'(y - \rho W y) \end{aligned} \tag{4.24}$$

by setting (4.24) equal to zero we get the result □

However, this estimate is biased since

$$\begin{aligned} \mathbb{E}(\hat{\rho}) &= \mathbb{E}(y'W'Wy)^{-1}y'W'(\rho W y + \epsilon) \\ &= \rho + \mathbb{E}(y'W'Wy)^{-1}y'W'\epsilon \end{aligned} \tag{4.25}$$

we cannot show that $\mathbb{E}(\hat{\rho}) = \rho$. That is different to the time series case, where Wy contains only past values of the process which are independent of the current value of the noise ϵ. In this case $\mathbb{E}\epsilon = 0$ implies unbiasedness.

Therefore the inappropriateness of the least squares estimator for models that incorporate spatial dependence has focused attention on the maximum likelihood approach as an alternative. Going back to the early work of Whittle [83] and Mead [64], maximum likelihood approaches have been suggested and derived for spatial autoregressions.

The maximum likelihood function for the (4.20) is shown as follows

$$L(y|\rho, \sigma^2) = \frac{1}{2\pi\sigma^{2(n/2)}}|I_n - \rho W| exp\{-\frac{1}{2\sigma^2}(y - \rho W y)'(y - \rho W y)\} \tag{4.26}$$

and by substituting $\hat{\sigma^2} = (1/n)(y - \tilde{\rho} W y)'(y - \tilde{\rho} W y)$ into eqn.(4.26) and taking logs will yields:

$$ln(L) \propto -\frac{n}{2}\log(y - \rho W y)'(y - \rho W y) + \log|I_n - \rho W| \tag{4.27}$$

This expression can be maximized with respect to ρ using a simple univariate optimization routine to get the ML-estimate $\tilde{\rho}$. Now, by substituting $\tilde{\rho}$ into the equation of $\hat{\sigma^2}$ we get the estimate of σ^2.

4.3.2 Review To The Bayesian Analysis

In this section we are going to give a brief introduction of the Bayesian analysis. We refer to Lee [58] for a complete introduction to Bayesian analysis and a good lecture notes can be found,e.g., Walsh in [81]

4.3.2.1 Bayes' Theorem

The foundation of Bayesian statistics is Bayes' Theorem. Suppose we observe random variable y and want to make inferences about another variable θ, where θ is drawn from some distribution $p(\theta)$, then from the definition of conditional probability,

$$Pr(\theta|y) = \frac{Pr(y,\theta)}{Pr(y)} \tag{4.28}$$

We can express the joint probability on θ to give

$$Pr(y,\theta) = Pr(y|\theta)Pr(\theta) \tag{4.29}$$

Putting together we get Bayes' theorem:

$$Pr(\theta|y) = \frac{Pr(y|\theta)Pr(\theta)}{Pr(y)}, \tag{4.30}$$

in particular if θ is discrete with n possible outcomes $(\theta_1, ..., \theta_n)$.

$$\Pr(\theta_j|y) = \frac{Pr(y|\theta_j)Pr(\theta_j)}{Pr(y)} = \frac{Pr(y|\theta_j}{\sum_{i=1}^{n} Pr(\theta_i)Pr(y|\theta_i)} \tag{4.31}$$

$Pr(\theta)$ is the **prior distribution** of the possible θ values, while $Pr(\theta|y)$ is the **posterior distribution** of θ given the observed data y

4.3.2.2 The Posterior Distribution

Generally the posterior distribution is obtained by simulation using Gibbs sampling, and hence the Bayes estimate of a parameter is frequently presented as a frequency histogram from (Gibbs) samples of the posterior distribution

4.3.2.3 The Choice of A Prior

The critical feature of Bayesian analysis is the choice of a prior, where the shape (family) of the prior distribution is often chosen to facilitate calculation of posterior, especially through the use of **conjugate priors** that, for a given likelihood function, return a posterior in the same distribution family as the prior (i.e., a gamma prior returning a gamma posterior when the likelihood is Poisson).

Definition 4.5. *: Diffuse Priors*
The most common priors is the flat, uninformative, or diffuse prior where the prior is simply a constant,

$$p(\theta) = k = \frac{1}{b-a} \text{ for } a \leq \theta \leq b \tag{4.32}$$

Definition 4.6. *: Jeffrey's Prior*
Jeffreys [50] proposed a general prior, based on the Fisher information I of the likelihood. Recall that

$$I(\theta|x) = -\mathbb{E}_x(\frac{\partial^2 \ln l(\theta|x)}{\partial \theta^2})$$

Jeffreys' rule(giving the Jeffreys'Prior) is to use as the prior

$$p(\theta) \propto \sqrt{I(\theta|z)} \tag{4.33}$$

A full discussion with derivation can be found in Lee [58], section 3.3.

When there are multiple parameters, $\mathbf{I}$ *is the Fisher Information matrix, the matrix of the expected second partials,*

$$I(\Theta|\mathbf{x})_{\mathbf{ij}} = -\mathbb{E}_{\mathbf{x}}(\frac{\partial^{\mathbf{2}} \ln \mathbf{l}(\mathbf{\Theta}|\mathbf{x})}{\partial \theta_{\mathbf{i}} \partial \theta_{\mathbf{j}}})$$

and the Jeffreys' Prior becomes

$$p(\Theta) \propto \sqrt{\det[\mathbf{I}(\theta|\mathbf{x})]} \tag{4.34}$$

4.3.2.4 Posterior Distribution Under Normality Assumptions

Consider the case where data are drawn from a normal distribution so that the likelihood function for the ith observation, x_i,

$$l\ (\mu, \sigma^2|x_i) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\Big(-\frac{(x_i-\mu)^2}{2\sigma^2}\Big) \tag{4.35a}$$

$$l\ (\mu|\mathbf{x}) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\Big(-\sum_{i=1}^{n} \frac{(x_i-\mu)^2}{2\sigma^2}\Big) \tag{4.35b}$$

$$= \frac{1}{\sqrt{2\pi\sigma^2}} \exp\Big[\Big(-\frac{1}{2\sigma^2}\sum_{i=1}^{n} x_i^2 - 2\mu n\bar{x} + n\mu^2\Big)\Big] \tag{4.35c}$$

where (4.35b) is the full likelihood for all n data points.

Known Variance and Unknown Mean

Lemma 4.2. *Assume the variance σ^2 is known, while the mean μ is unknown. Then it remains to specify the prior for $\mu, p(\mu)$. Suppose we assume a Gaussian prior,i.e., $\mu \sim \mathcal{N}(\mu_0, \sigma_0^2)$, so that*

$$p(\mu) = \frac{1}{\sqrt{2\pi\sigma_0^2}} \exp\Big(-\frac{(\mu-\mu_0)^2)}{2\sigma_0^2}\Big) \tag{4.36}$$

Then the posterior density function for μ is a normal with mean μ_* and variance σ_*^2, e.g.,

$$\mu|(x, \sigma^2) \sim \mathcal{N}(\mu_*, \sigma_*^2) \tag{4.37}$$

where

$$\sigma_*^2 = \Big(\frac{1}{\sigma_0^2} + \frac{n}{\sigma^2}\Big) \quad \textit{and} \quad \mu_* = \sigma_*^2\Big(\frac{\mu_0}{\sigma_0^2} + \frac{n\bar{x}}{\sigma^2}\Big)$$

The mean and variance of the prior, μ_0 and σ_0 are referred to as **hyperparameters**. Note that in the Bayesian analysis we can ignore known parameters and treat them as constants, i.e., suppose $\mathbf{x}$ denotes the data, and Θ_1 is a vector of known model parameters, while Θ_2 is a vector of unknown parameters. If we can write the posterior as

$$p(\Theta_2|\mathbf{x}, \Theta_1) = f(\mathbf{x}, \Theta_1).g(x, \Theta_1, \Theta_2) \tag{4.38a}$$

$$\text{then} \quad p(\Theta_2|\mathbf{x}, \Theta_1) \propto g(\mathbf{x}, \Theta_1, \Theta_2) \tag{4.38b}$$

Before we examine a Gaussian likelihood with unknown variance, we need to develop χ^{-2}, the **inverse chi-square distribution** via gamma and inverse-gamma distribution.

Definition 4.7. *:Gamma,Inverse-Gamma, χ^2, and χ^{-2}Distributions*

- *A gamma-distributed variable is denoted by* $x \sim Gamma(\alpha, \beta)$*, with density function*

$$p(x|\alpha, \beta) = \frac{\beta^\alpha}{\Gamma(\alpha)} x^{\alpha-1} e^{-\beta x} \quad for \;\; \alpha, \beta, x > 0$$

As a function of x*, note that*

$$p(x) \;\; \propto x^{\alpha-1} e^{-\beta x}$$

The mean and variance of this distribution are

$$\mu_x = \frac{\alpha}{\beta}, \quad \sigma_x^2 = \frac{\alpha}{\beta^2}$$

$\Gamma(\alpha)$*, is a Gamma function evaluated at* α*,i.e.,*

$$\Gamma(\alpha) = \int_0^\infty y^{\alpha-1} e^{-y} dy$$

where

$$\Gamma(\alpha + 1) = \alpha\Gamma(\alpha) = \alpha! = \alpha(\alpha - 1)!,$$
$$\Gamma(1) = 1, \quad \Gamma(1/2) = \sqrt{\pi}$$

Reference for the Gamma function can be found, e.g.,Abramowitz and Stegun [1].

- *The* χ^2 *distribution is a special case of gamma, for a* χ^2 *with* n *degrees of freedom is a gamma random variable with* $\alpha = n/2, \beta = 1/2$*, i.e.,* $\chi_n^2 \sim Gamma(n/2, 1/2)$*, and has density function as*

$$p(x|n) = \frac{2^{-n/2}}{\Gamma(n/2)} x^{n/2-1} e^{-x/2}$$

Hence for a χ_n^2*,*

$$p(x) \propto x^{n/2-1} e^{-x/2}$$

- *The inverse gamma distribution is defined by the distribution of* $y = 1/x$ *where* $x \sim Gamma(\alpha, \beta)$*. The density function, mean and variance are*

$$p(x|\alpha, \beta) = \frac{\beta^\alpha}{\Gamma(\alpha)} x^{-(\alpha-1)} e^{-\beta/x} \quad for \;\; \alpha, \beta, x > 0$$
$$\mu_x = \frac{\beta}{\alpha - 1}, \quad \sigma_x^2 = \frac{\beta^2}{(\alpha - 1)^2(\alpha - 2)}$$

Note for the inverse gamma that

$$p(x) \propto x^{-(\alpha-1)} e^{\beta/x}$$

- *Similarly, if* $y = 1/x$ *and* $x \sim \chi_n^2$ *then* y *will follow an inverse chi-square distribution denoted by* $y \sim \chi_n^{-2}$. *As in the* χ^2 *distribution then the inverse* χ^2 *distribution is also a special case of the inverse gamma, with* $\alpha = n/2, \beta = 1/2$. *The density function, mean and variance are*

$$p(x|n) = \frac{2^{-n/2}}{\Gamma(n/2)} x^{-(n/2-1)} e^{-1/(2x)}$$
$$\mu_x = \frac{1}{n-2}, \quad \sigma_x^2 = \frac{2}{(n-2)^2(n-4)}$$

- *The scale inverse chi-square distribution is defined as*

$$p(x|n) \propto x^{-(n/2-1)} e^{-\sigma_0^2/(2x)}$$

so that the $1/(2x)$ *term in the exponential is replaced by an* $\sigma_0^2/(2x)$ *term. This distribution is denoted by* $SI - \chi^2(n, \sigma_0^2)$ *or* $\chi^{-2}_{(n,\sigma_0^2)}$.
Note that if $x \sim \chi^{-2}_{(n,\sigma_0^2)}$ *then* $\sigma_0^2 x \sim \chi^{-2}$

Unknown Variance: Inverse-χ^2 Priors

Lemma 4.3. *Now suppose the data are drawn from a normal with known mean* μ, *but unknown variance* σ^2. *The resulting likelihood function becomes*

$$l(\sigma^2|\mathbf{x}, \mu) \propto (\sigma^2)^{-n/2}. \exp\Big(-\frac{S^2}{2\sigma^2}\Big) \tag{4.39a}$$

$$\text{where} \quad S^2 = \sum_{i=1}^{n} (x_i - \mu)^2 \tag{4.39b}$$

In principle we might have any form of prior distribution for the variance σ^2. *However, if we are to be able to deal easily with the posterior distribution it helps if the posterior distribution is of a 'nice' form. This will certainly happen if the prior is of a similar form of the likelihood, namely*

$$p(\sigma^2) \sim (\sigma^2)^{-\kappa/2} \exp\Big(-\frac{\sigma_0^2}{2\sigma^2}\Big)$$

where κ *and* σ_0^2 *are suitable constants. For the reason of getting 'nice' form or the posterior, we can substitude* $\kappa = \nu_0 + 2$ *and the prior becomes*

$$p(\sigma^2) \sim (\sigma^2)^{-\nu_0/2-1}.\exp\Big(-\frac{\sigma_0^2}{2\sigma^2}\Big)$$

Then by multiplying prior to the likelihood we get the posterior as follow

$$\begin{aligned} p(\sigma^2|\mathbf{x},\mu) &\propto (\sigma^2)^{-n/2}\exp\Big(-\frac{S^2}{2\sigma^2}\Big)(\sigma^2)^{-\nu_0/2-1}.\exp\Big(-\frac{\sigma_0^2}{2\sigma^2}\Big) \\ &= (\sigma^2)^{-(n+\nu_0)/2-1}\exp\Big(-\frac{S^2+\sigma_0^2}{2\sigma^2}\Big). \end{aligned} \tag{4.40}$$

i.e. the posterior has inverse chi-square distribution.

Proof. can be seen in Lee [58] □

Unknown Mean and Variance

It is realistic to suppose that both parameters of a normal distribution are unknown rather than just one.

Lemma 4.4. *Now we consider* $\mathbf{x} = (x_1, x_2, ..., x_n)$ *as our observations, which are* $\mathcal{N}(\mu, \sigma^2)$ *with* μ *and* σ^2 *unknown. Clearly,*

$$\begin{aligned} p(\mathbf{x}|\mu,\sigma^2) &= (2\pi\sigma^2)^{-\frac{1}{2}}\exp\Big(-\frac{(x-\mu)^2}{2\sigma^2}\Big) \\ &= \{(2\pi)^{-\frac{1}{2}}\}\{(\sigma^2)^{-\frac{1}{2}}\}\exp\Big(-\frac{\mu^2}{2\sigma^2}\Big)\exp\Big(\frac{x\mu}{\sigma^2}-\frac{x^2}{2\sigma^2}\Big) \end{aligned}$$

from which it follows that the density is in the two-parameter exponential family. Further the likelihood

$$\begin{aligned} l\ \ (\mu,\sigma^2|\mathbf{x}) &\propto p(\mathbf{x}|\mu,\sigma^2) \\ &\propto (\sigma^2)^{-n/2}\exp\Big[-\frac{1}{2}\sum(x_i-\mu)^2/\sigma^2\Big] \\ &= (\sigma^2)^{-n/2}\exp\Big[-\frac{1}{2}(S+n(\bar{x}-\mu)^2)/\sigma^2\Big] \end{aligned}$$

where $S = \sum(x_i - \bar{x})^2$

This case can get complicated. For a brief introduction we consider the case of an indifference or 'reference' prior. It is usual to take

$$p(\mu, \sigma^2) \propto 1/\sigma^2$$

which is a product of reference prior $p(\mu) \propto 1$ *for* μ *and* $p(\sigma^2) \propto 1/\sigma^2$ *for* σ^2*. Then the posterior will be*

$$p(\mu, \sigma^2|\mathbf{x}) \propto (\sigma^2)^{-n/2-1} \exp\Big[-\frac{1}{2}(S + n(\bar{x} - \mu)^2)/\sigma^2\Big]$$

Proof. See Lee [58], p.73-75. □

4.3.3 Bayesian Spatial Autoregressive Model

Now we combine the spatial autoregressive model and the Bayesian technique. Assume that $p(\rho, \beta, \sigma, V) = p(\rho)p(\beta)p(\sigma)p(V)$; that is, the priors are independent. The models and prior information of Bayesian spatial autoregressive model are shown as follow:

$$\begin{aligned} y &= \rho \cdot Wy + Z\beta + \epsilon \\ \epsilon &\sim \mathcal{N}(0, \sigma^2 \mathrm{V}) \\ V &= diag(v_1, v_2, ..., v_n) \\ \beta &\sim \mathcal{N}(\bar{\beta}, \mathrm{Var}(\beta)) \\ \sigma^2 &\sim 1/\sigma \\ \rho &\sim constant \\ r/v_i &\sim ID\chi^2(r)/r \\ r &\sim constant \end{aligned} \tag{4.41}$$

Where y is an nx1 vector of dependent variables and Z represents the nx k matrix of explanatory variables. The W is an nx n matrix representing the spatial weight matrix. Assume that ϵ is an nx1 vector of normal distribution random variates with non-constant variance. We use normal prior with hyperparameters $\bar{\beta}, Var(\beta)$ on parameters β and

diffuse prior for σ^2. Prior for ρ is constant. However, we can also choose uniform distribution with hyperparameters r_{min}, r_{max} for ρ. We use $(-1, 1)$ as a default for r_{min}, r_{max}. Another alternative is using Beta distribution with hyperparameters a_1, a_2 for ρ, particularly we choose $a_1 = a_2 = 1.01$ as our default. The relative variance terms $(v_1, v_2, ..., v_n)$ are assumed fixed but unknown parameters that need to be estimated. The thought of estimating n parameters $v_1, v_2, ..., v_n$, in addition to the $2k+1$ parameters, β, ρ, σ using n data observations seems problematic. However, by using Bayesian methods that problem goes away, because we can rely on an informative prior for these parameters. This prior distribution for the v_i terms will take the form of an independent $\chi^2(r)/r$ distribution, with r is the χ^2's parameter. The prior of r is again a constant. However, Gamma distribution with hyperparameter m, k, can also be considered as a choice of prior distribution. This type of prior has been used by Geweke [35] in modelling heteroscedasticity and outliers.

The specifics regarding the prior assigned to the v_i terms can be motivated by considering that the prior mean equals unity and the variance of the prior is $2/r$. This implies that as r becomes very large the terms v_i will be approach to unity, resulting in $V = I_n$, the tradiotional Gauss-Markov assumption. Large r values are associated with a prior belief that outliers and non-constant variances do not exist.

Theorem 4.1. *The posterior density kernel for the models is the product of the kernel densities of the independent prior distributions shown in (4.42), and the likelihood function will be*

$$\begin{aligned} L(\rho, \beta, \sigma^2, v; yW) &= \sigma^{-1}|I_n - \rho W| \prod_{i=1}^{n} v_i^{-1/2} \exp\Big[-\sum_{i=1}^{n} (\epsilon_i^2/2\sigma^2 v_i)\Big] \\ &= \sigma^{-1} \prod_{i=1}^{n} (1 - \rho\lambda_i) \prod_{i=1}^{n} v_i^{-1/2} \exp\Big[-\sum_{i=1}^{n} (\epsilon_i^2/2\sigma^2 v_i)\Big] \end{aligned} \tag{4.42}$$

where ϵ_i is the i^{th} element of the vector $(y - \rho Wy - Z\beta)$. The λ_i denoting the eigenvalues of the spatial weight matrix W. This gives the posterior density kernel

$$p(\rho,\beta,\sigma,V) \propto \prod_{i=1}^{n}(1-\rho\lambda_i)\prod_{i=1}^{n} v_i^{-(r+3)/2}\exp(-r/2v_i) \tag{4.43}$$

$$\cdot\,\sigma^{-(n+1)}\exp\Big[\sum_{i=1}^{n}(\sigma^{-2}\epsilon_i^2+r)/2v_i\Big] \tag{4.44}$$

Proof. see Geweke [35] □

To bring this model into Gibbs sampler, first we consider the conditional posterior for σ given ρ,β and $v_1,v_2,...,v_n$ then (4.43) will be left to be only (4.44), i.e.,

$$p(\sigma|\rho,\beta,V) \propto \sigma^{-(n+1)}\exp\Big[\sum_{i=1}^{n}(\sigma^{-2}\epsilon_i^2+r)/2v_i\Big]$$

Geweke [35] shows that this result in a conditional $\chi^2(n)$ distribution for σ as follow

$$\sum_{i=1}^{n}(\epsilon_i^2/v_i)/\sigma^2 \sim \chi^2(n) \tag{4.45}$$

The conditional distribution β takes the standard multivariate normal with mean and variance

$$\bar{\beta} = (\tilde{Z}'V^{-1}\tilde{Z})^{-1}\tilde{Z}'V^{-1}\tilde{y} \tag{4.46}$$

$$var(\beta) = \sigma^2(\tilde{Z}'V^{-1}\tilde{Z})^{-1} \tag{4.47}$$

$$\tilde{Z} = (I_n-\rho W)Z$$

$$\tilde{y} = (I_n-\rho W)y$$

Similarly, for the posterior distribution of $v_1,v_2,...,v_n$, conditional on ρ,β,σ^2 we can follow Geweke [35] and find that

$$(\sigma^{-2}\epsilon_i^2+r)/v_i \sim \chi^2(r+1) \tag{4.48}$$

Now, the condistional posterior distribution for ρ, the spatial autocorrelation parameter, conditioning on σ, β and $v_1, v_2, ..., v_n$, can be shown as

$$p(\rho|\beta, \sigma, V) \propto |I_n - \rho W| \exp\Big[-(1/2\sigma^2)(\epsilon' V^{-1}\epsilon)\Big] \tag{4.49}$$

Given those conditional posterior densities in (4.45) through (4.49), we can formulate a Gibbs sampler for this model using the following steps (LeSage [51]):

1. Begin with arbitrary values for the parameters $\sigma^0, \beta^0, \rho^0$ and v_i^0
2. Compute (4.45) using ρ^0, β^0 and v_i^0, and use it along with a random $\chi^2(n)$ draw to determine σ^1.
3. Determine the means and variances for β using (4.46) and (4.47). Carry out a multivariate random draw based on this mean and variance to determine β^1.
4. Using σ^1, β^1 and ρ^0, calculate (4.48) and use it with an $n-$vector of random $\chi^2(r+1)$ draws to determine $v_i^1, i = 1, ..., n$.
5. Use ratio of uniforms sampling to determine ρ^1 from (4.49) given the values σ^1, β^1 and $v_i^1, i = 1, ..., n$

These steps constitute a single pass of the Gibbs sampler, where initial arbitrary values of β^0, ρ^0, v_i^0 and σ^0 have been replaced with j values of β^j, ρ^j, v_i^j and $\sigma^j, j = 1, ..., M$, e.g., $M = 1000$ from which we can approximate the posterior distribution for the parameters.

4.3.4 The Application into Image correction

We are going to model the error of the residual image to reduce the blur effect that we get from the two dimensional bootstrap synthesize using (4.42).

Let $Y(i,j) = X(i,j) - X^*(i,j)$, where $i = 1, ..., m\hat{T}_1$ and $j = 1, ..., n\hat{T}_2$, $m\hat{T}_1 \leq M, n\hat{T}_2 \leq N$ is the error of the bootstrap to the original image. We arrange the image data $Y(i,j)$ as y, a $(m\hat{T}_1 \cdot n\hat{T}_2)$x 1 vector of dependent variables. The vector y will be $y = [Y(1,1), ..., Y(m\hat{T}_1, 1), Y(1,2), ..., Y(m\hat{T}_1, 2), ..., Y(1, n\hat{T}_2), ..., Y(m\hat{T}_1, n\hat{T}_2)]$. As the

1	6	11	16	21
2	7	12	17	22
3	8	13	18	23
4	9	14	19	24
5	10	15	20	25

Figure 4.12: reindexing the matrix for 5 x 5 matrix into 25 x 1 vector

consequence of this arrangement, then our index of will run from $i = 1, ..., m\hat{T}_1 \cdot n\hat{T}_2$ rowise.

The explanatory variable Z, in our case will be the index from the neighborhood pixel. This depends on the assumption that we make; either we assume it has a toroidal or not a toroidal boundary. The order of the neighborhood gives the number of columns in the matrix Z.

For toroidal boundary, order of neighborhood two, and the size of image as in Figure.4.12, we have

$$Z = \begin{pmatrix} 5 & 2 & 21 & 26 \\ 1 & 3 & 22 & 7 \\ . & . & . & . \\ 10 & 7 & 1 & 11 \\ 6 & 8 & 2 & 12 \\ . & . & . & . \\ . & . & . & . \\ 24 & 21 & 20 & 5 \end{pmatrix}$$

and the size of Z will be $(m\hat{T}_1 \cdot n\hat{T}_2)$x k. In case we consider no toroidal boundary, we have to reduce the error image Y. We exclude the boundary of Y such that no neighborhood of a pixel will lie outside of the error image.

For the size of image as in Figure.4.12, without toroidal boundary, the explanatory variabel Z will be

$$Z = \begin{pmatrix} 6 & 8 & 2 & 12 \\ 7 & 9 & 3 & 13 \\ . & . & . & . \\ . & . & . & . \\ 17 & 19 & 13 & 23 \\ 18 & 20 & 14 & 24 \end{pmatrix}$$

Next, we need to set the spatial weighted matrix,W. There are many posibilities to define this matrix, e.g. Anselin [4], but particularly for this application we choose

$$W(s,t) = \begin{cases} 1/k, & \text{if } < s,t > \text{is the neighbour of (i,j);} \\ 0, & \text{otherwise} \end{cases}$$

where $i, j = 1, ..., m\hat{T}_1 \cdot n\hat{T}_2$.

The matrix W is a sparse matrix. To avoid huge storage in the computational, we only need to save the index of the neighbourhood,e.g., W_{index} is a $m\hat{T}_1 \cdot n\hat{T}_2$ x k. We set $W(i,j) = 1/k$ but now $i = 1, ..., m\hat{T}_1 \cdot n\hat{T}_2$ and $j = W_{index}(1,1), ..., W_{index}(m\hat{T}_1 \cdot n\hat{T}_2, k)$ and set W as a sparse matrix.

We use the Matlab Toolbox for spatial statistics from James LeSage [59] for the Bayesian SAR computation. We transfer back $\hat{y}$, the $m\hat{T}_1 \cdot n\hat{T}_2, k$ x 1 estimation vector of Y by Bayesian SAR, into $\hat{Y}$ a $m\hat{T}_1$ x $n \cdot \hat{T}_2, k$ matrix. Finally, we get the correction image as $\tilde{X} = \hat{X} + X^*$.

As an illustration, we use this procedure to reduce the error in the Figure.4.12. We use 128x128 image as our sample, in Figure.4.13A. The bootstrap image is in Figure.4.13B, and the error correction with is in Figure.4.13C. The image of error correction is produced by adding gray value of Figure.4.13B, i.e. the Bootrap image, to the Error prediction by Bayesian SAR (Figure.4.13B).

The variance of residual from Bayesian SAR is depicted in Figure.4.14 and the partial comparasion between bootstrap error, i.e., gray values of Figure.4.13A- gray values of Figure.4.13B, and the prediction to Bayesian SAR is depicted in Figure.4.15.

Figure 4.14 shows a plot of the mean of the variance, v_i, draws, conforming that handful of large v_i values exist. This figure is only a part of the whole plot, the total amounts of v_i for 128x128 image (excluding the border) will be 15876. The next Figure.4.15 is a

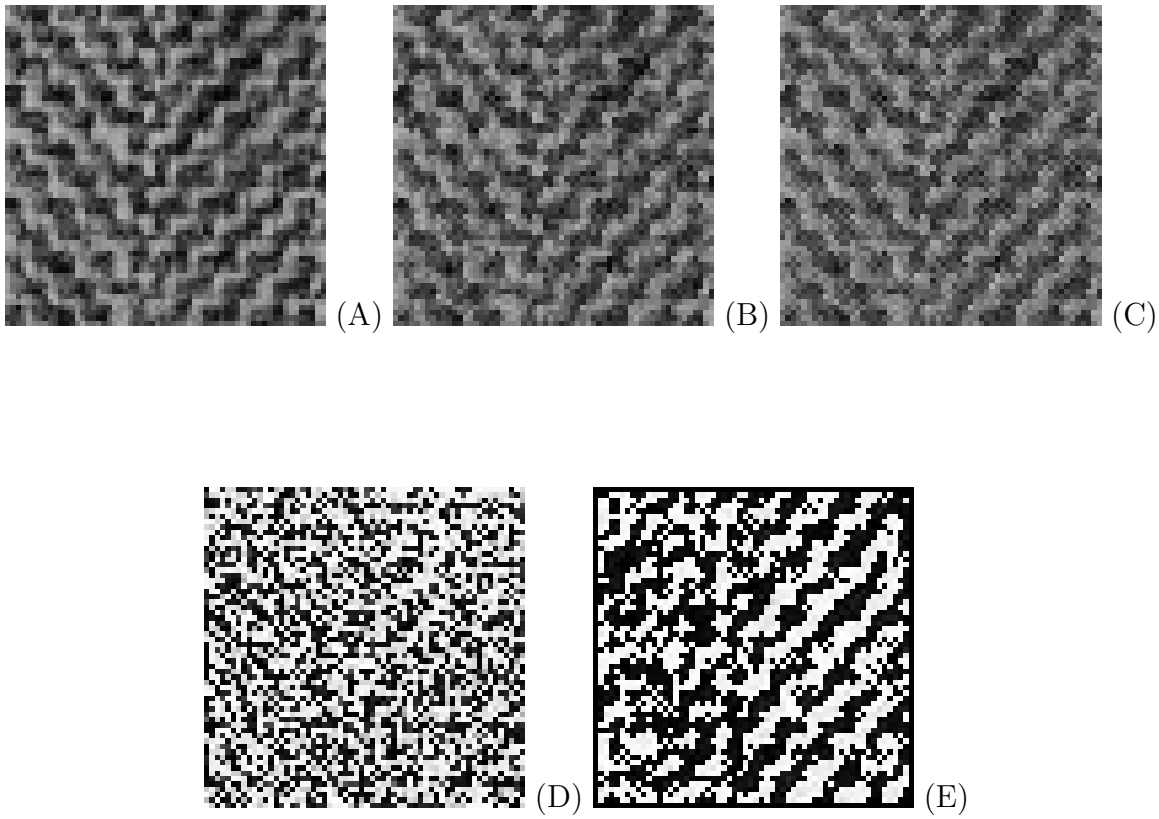


Figure 4.13: (A). original image, (B). Synthesis by Bootstrap, (C).The Error correction with Bayesian SAR, (D). Error between the original image and the bootstrap, (E). The Error prediction by Bayesian SAR

plot of the residual from bootrap image to the original vs the prediction of this residual from Bayesian SAR. We can see that the prediction can follow the dynamic of the 'true' residual, but visually, the error is still relatively large.

4.4 Correction in the edges between two consecutive images

In these two texture synthesize methods, i.e., using Markov random field and two dimensional bootstrap. Due to the assumption of continous image sometimes we run into problems in two consecutive sample images for performing a larger image.

Efros and Freeman [24] have an idea that the process of texture synthesis would be akin to putting together a jigsaw puzzle, quilting together the patches, making sure they all fit together.

We want to make the cut between two overlapping blocks on the pixels where the edges between these consecutive images meet. This can be done with dynamic programming. The minimal cost path through the error surface is computed in the following manner.

Let B_1 and B_2 be two blocks that overlap along their vertical edge with the regions of overlap B_1^{ov} and B_2^{ov} respectively, then the error surface is defined as $e = (B_1^{ov} - B_2^{ov})^2$. To find the minimal vertical cut through this surface we traverse $e(i = 2..N)$ and compute the cumulative minimum error E for all paths:

$$E_{ij} = e_{ij} + \min(E_{i-1j-1}, E_{i-1j}, E_{i-1j+1}) \tag{4.50}$$

In the end, the minimum value of the last row in E will indicate the end of the minimal vertical path through the surface and we can trace back and find the path of the best cut. Similar procedure can be applied to horizontal overlaps.

Figure.4.16 give examples of Efros and Freeman's procedure applying to the simulated images from the bootstrap method. We can see there are some break lines in the horizontal (vertical) in the figures of the left part. These break lines are the edges between two consecutive sample images. By using the quilting method these errors can be removed and the simulated images look nice.

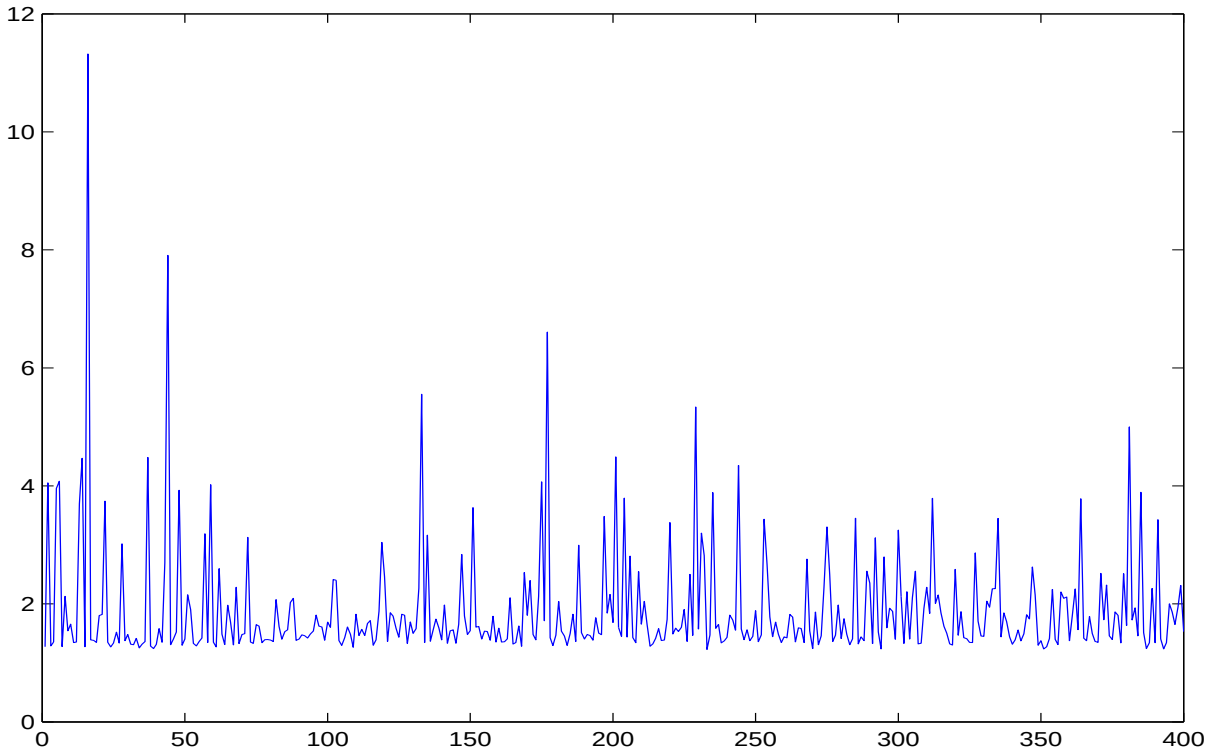


Figure 4.14: Part of the Variance Estimate of Residual from Bayesian SAR

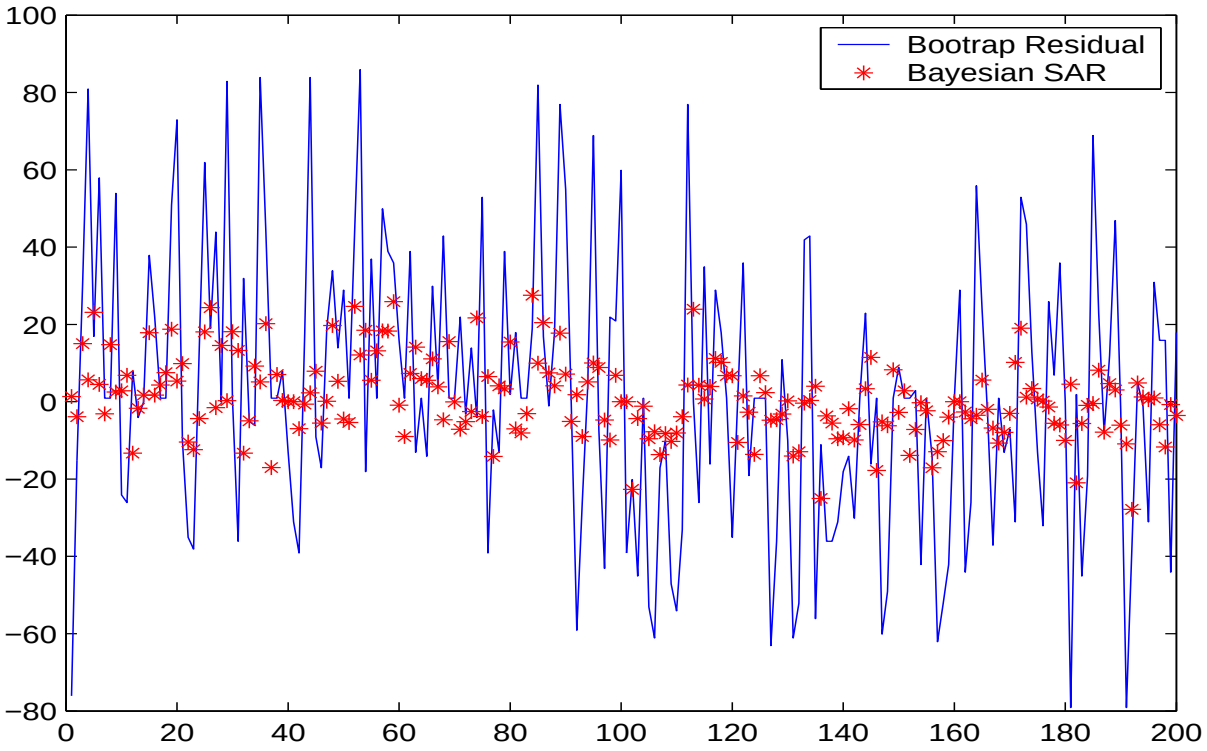


Figure 4.15: The Bootstrap Error Vs The Prediction of Bayesian SAR

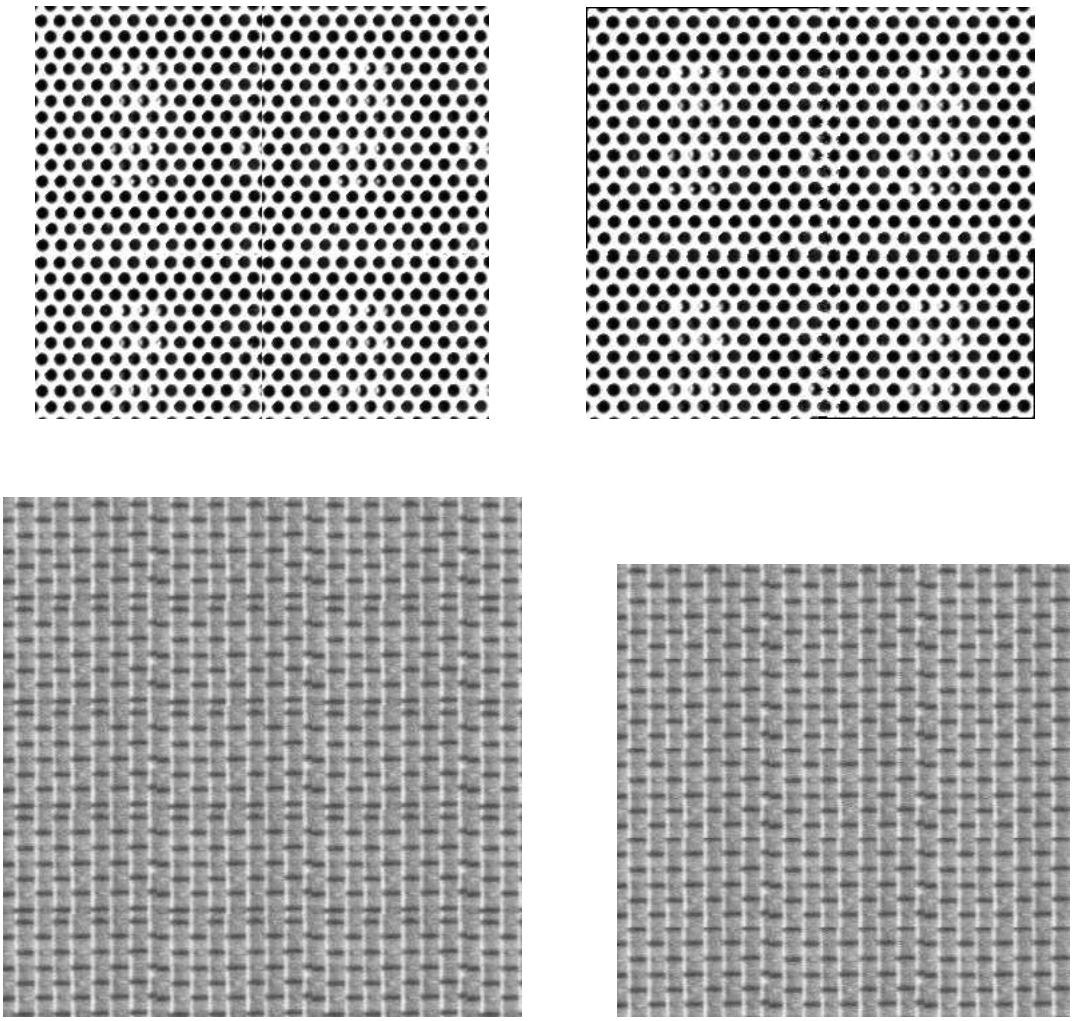


Figure 4.16: Examples of error in the edges between two consecutive images and the correction ones

Chapter 5

Defect Detection on Texture

In the industrial problem defect detection plays a major role in quality control. So far quality control in many applications takes place manually, by checking the defect using random sampling and performing a quality control chart. The results of this manual procedure is of course dependant on the examiner. Even as it is stated in Chetverikov [13], that humans have the capability to easily find imperfections in spatial structures, they have physical fatigue that reduces the performance of defect detection. An automatic inspection system replaces human eyes with cameras, part of their brains by computers, and part of their abilities to detect the error by software. But we, the humans, have to give a good procedure for detecting the defect by analyzing the images that are produced by cameras. Some applications for automatic defect detection are summarized in Kohrt [55].

In this work we consider only defect detection in the texture. Numerous methods have been designed to solve particular texture inspection tasks. Cohen,et.al [76] used MRF models for defect inspection in textile surface, Chetverikov [14] using regularity and local orientation for finding defects in texture. Meanwhile Sezer, et.al [77] use Independent Component Analysis for the same purpose.

We treat defect detection as a multihypothesis testing problem with the null hypothesis representing the absence of defects and the alternative hypotheses representing various types of defects. Departure from the undisturbed surface models that we generated in previous chapter, i.e., texture synthesis and various summary statistics are going to be investigated which represent the regularity of the surface in a given part of the observation

area. They are going to be used to test the null hypothesis of absence of defects in the particular segment against the alternatives representing different types of defects.

5.1 The Hypothesis Analysis Construction

We assume that we have a pair of images with the same size; one is an image without defect, the other is the defect one. The defect detection of this image is achieved by comparing the two series that are taken from the line, rowwise and columnwise, in the same position in the image.

Several studies of methods for comparing nonparametric versus parametric regression fits have been done, for example in Hall and Hart [41], King, et.al [54], Härdle and Mammen [43], Delgado [22], compare also Härdle, et.al [44] for testing the parametric versus semiparameric model and Neumeyer [66] for comparision of regression curves, when the error is heteroscedastics. Hart [45] gives a complete study through the business of nonparametric smoothing and lack-of-fit tests. In this book, he gives an introduction to some nonparametric methods of function estimation, and shows how they can used to test the adequacy of parametric function estimates. The lack of fit tests are explained from the classical ones, and also based on linear smoothers. Furthermore, the extention for comparing curves is also given as a part the extending the scope of application.

We follow the idea of these studies, but instead of comparing different models with one series we compare two series with one model.

An introduction to some nonparametric methods of function estimation, and showing how they can be used to test the adequacy of parametric function estimates is fiven in Hart [45]. In this book Hart explains the Lack-of Fit tests based on Linear Smoothers as a main part. Furthermore, the extension for comparing curves is also given as a part of extending the scope of application.

We first consider the following nonparametric regression setup

$$Y_i = m^I(x_i) + \epsilon_i, \quad \tilde{Y}_i = m^{II}(x_i) + \tilde{\epsilon}_i, \quad i = 1, ..., n \tag{5.1}$$

where the $\epsilon_1, ..., \epsilon_n$, $\tilde{\epsilon}_1, ...\tilde{\epsilon}_n$ are independent with mean zero and finite variance, $Var(\epsilon_i) = Var(\tilde{\epsilon}_i) = \sigma^2(x_i)$ and uniformly bounded fourth moments $\mathbb{E}\epsilon_i^4, \mathbb{E}\tilde{\epsilon}_i^4 \leq C < \infty, i = 1, ..., n$.

For sake of simplicity, we only consider the case of equidistant x_i on a compact set, say $[0, 1]$.

We wish to test

$$H_0 : m^I(x_i) = m^{II}(x_i) = m(x_i), i = 1, ..., n, \text{ against}$$
$$H_1 : m^I(x_i) \neq m^{II}(x_i), \quad \text{for some } i.$$

We estimate m^I, m^{II} by μ^I, μ^{II} respectively using Priestley-Chao estimation.

$$\mu_h^I(x) = \frac{1}{n}\sum_{i=1}^{n} K_h(x - x_i)Y_i \tag{5.2a}$$

$$\mu_h^{II}(x) = \frac{1}{n}\sum_{i=1}^{n} K_h(x - x_i)\tilde{Y}_i \tag{5.2b}$$

where $K_h(.)$ denotes $h^{-1}K(./h)$ for a kernel K.

To perform a test, first we need to measure the distance between $\mu_h^I(x)$ and $\mu_h^{II}(x)$ and use this distance as test statistic for testing the null hypothesis. Following, Härdle and Mammen [43], we use standardized L_2-distance between these two estimates,i.e.

$$T_n = n\sqrt{h}\int \left(\mu_h^{II}(x) - \mu_h^I(x)\right)^2 dx \tag{5.3}$$

5.2 Assumptions

We use similar assumptions that we have already used in the chapter 2. However, for the convenience of reading, we are going to state them again in this section.

(A1). $m^I(.), m^{II}(.)$ are twice continuously differentiable.

(A2). $\sigma^2(x) = Var(Y_i|X_i = x)$ is bounded away from 0 and from ∞, uniformly in x, and it satisfies a Lipschitz condition.

(A3). m^{II} can be written as $m^{II}(x) = m^{I}(x) + c_n(x)\Delta_n(x)$ with $c_n = (n\sqrt{h})^{-1/2}$ and $\Delta_n(x)$ bounded uniformly in x and n (H_0 corresponds to $\Delta_n(.) \equiv 0$).

For the kernel K we use the following assumptions

(K1). The kernel K is a symmetric, twice continuously differentiable function with compact support $[-1, 1]$, furthermore $\int K(u)du = 1, \int uK(u)du = 0$.

(K2). The bandwidth h fulfills $h = h_n \sim cn^{-1/5}$ for some $c > 0$.

As we shall use the following approximation result for sums by integrals repeatedly, we formulated it there as a Lemma.

Lemma 5.1. *For any Lipschitz function g on $[a, b]$*

$$\left| \int_a^b g(x)dx - \sum_{j=1}^{n} g(x_j - 1)\Delta x_j \right| = O(\frac{1}{n}) \tag{5.4}$$

for $\Delta x_j = x_j - x_{j-1} = \frac{b-a}{n}, \quad j = 1, 2, ..., n.$ $x_0 = a, x_n = b, i.e., x_j = a + \frac{b-a}{n}j,$ $j = 0, 1, 2, ..., n$

Proof. The left hand side of the (5.4) is

$$\begin{aligned}\left| \sum_{j=1}^{n} \int_{x_{j-1}}^{x_j} [g(x) - g(x_j)]dx \right| &\leq \sum_{j=1}^{n} \int_{x_{j-1}}^{x_j} |g(x) - g(x_j)|dx \\ &\leq L \sum_{j=1}^{n} \int_{x_{j-1}}^{x_j} |x - x_j|dx \\ &\leq L(b-a)\frac{1}{n}\end{aligned}$$

the second to last equation comes from the Lipschitz continuity property, and we use $|x - x_j| \leq \frac{1}{n}$ for $x_{j-1} \leq x \leq x_j$ □

5.3 Asymptotic property of T_n

In the first proposition of this chapter we will approximate the distribution of T_n by a Gaussian distribution. We measure the distance between these distributions by the following modification of the Mallows distance, which is also used by Härdle and Mammen in [43]

$$d(\mu,\nu) = \inf_{X,Y}(\mathbb{E}\|X-Y\|^2 \wedge 1 : \mathcal{L}(\mathrm{X}) = \mu, \mathcal{L}(\mathrm{Y}) = \nu)$$

convergence in this distance is equivalent to weak convergence.

We also use the following notation for convolution of a function g with the kernel K_h respectively with itself:

$$\begin{aligned} \mathfrak{K}_h g(x) &= \int K_h(x-u)g(u)du, \\ g^{(2)}(x) &= \int g(x-u)g(u)du, \\ g^{(4)}(x) &= \int g^{(2)}(x-u)g^{(2)}(u)du \end{aligned}$$

For later reference we state the following properties where the first follows immediately and the second follows from the fact that $g^{(2)}$ is the probability density of $U+V$ are i.i.d with probability density g.

Lemma 5.2. .

a) If g is symmetric, then $g^{(2)}$ is symmetric too and $g^{(2)}(x-y) = \int g(x-u)g(y-u)du$

b) if g is nonnegative and $\int g(u)du = 1$, then $\int g^{(2)}(x)dx = 1$

Proposition 5.1. *Assume (A1)-(A3), (K1) and (K2). Then,*

$$d\Big(\mathcal{L}(\mathrm{T_n}), \mathcal{N}(\mathrm{B_h}, \mathrm{V})\Big) \to 0$$

where

$$\begin{aligned} B_h &= B_h^1 + B_h^0, \\ B_h^1 &= \int (\mathfrak{K}_h \Delta_n(x))^2 dx \\ B_h^0 &= \frac{2}{\sqrt{h}} \int \sigma^2(x) dx \int K^2(u) du \\ V &= 8 \int \sigma^4(x) dx K^{(4)}(0). \end{aligned}$$

In particular, $B_h^1 \geq 0$ *and, under the hypothesis* H_0, $B_h^1 = 0$

Proof. The proof of this proposition is along the same lines as in Härdle and Mammen [43]. Using

$$m^{II}(.) = m^I(.) + c_n \Delta_n(.)$$

and we defining $\dot{\epsilon}_i = \tilde{\epsilon}_i - \epsilon_i$ for $i = 1, ..., n$, we get

$$\begin{aligned} T_n &= n\sqrt{h} \int (\mu_h^{II}(x) - \mu_h^I(x))^2 dx \\ &= n\sqrt{h} \int \Big[\frac{1}{n} \sum_{i=1}^n K_h(x_i - x) \{ m^{II}(x_i) + \tilde{\epsilon}_i - m^I(x_i) - \epsilon_i \} \Big]^2 dx \\ &= \frac{\sqrt{h}}{n} \int \sum_{i=1}^n \Big[K_h(x_i - x) \{ (m^{II}(x_i) - m^I(x_i)) + (\tilde{\epsilon}_i - \epsilon_i) \} \Big]^2 dx \\ &= \frac{\sqrt{h}}{n} \int \sum_{i=1}^n \Big[K_h(x_i - x) \{ c_n \Delta_n(x_i) + \dot{\epsilon}_i \} \Big]^2 dx \\ &= \frac{\sqrt{h}}{n} \int \Big[\sum_{i=1}^n K_h(X_i - x) c_n \Delta_n(x_i) + \sum_{i=1}^n K_h(x_i - x) \dot{\epsilon}_i \Big]^2 dx \\ &= \frac{\sqrt{h}}{n} \int \Big[U_{n,1}(x) + U_{n,2}(x) \Big]^2 dx \end{aligned}$$

Where we have set

$$\begin{aligned} U_{n,1}(x) &= \sum_{i=1}^n K_h(x_i - x) c_n \Delta_n(x_i) \\ U_{n,2}(x) &= \sum_{i=1}^n K_h(x_i - x) \dot{\epsilon}_i \end{aligned}$$

a.) First, we investigate the asymptotic behaviour of $U_{n,1}(x)$. By our smoothness assumptions on K, m^I, m^{II}, we have that K_h and Δ_n are Lipschitz continuous with Lipschitz constants of order h^{-2} and c_n^{-1} respectively. Therefore, $K_h(u-x)\Delta_n(u)$ is Lipschitz continuous in u with a constant of order $\max(h^{-2}, (hc_n)^{-1})$, as K, Δ_n are bounded, and by Lemma 5.1 the approximation error of the sum by the integral is of order

$$\frac{1}{n}\max(h^{-2}, (hc_n)^{-1}) = \max\Big(\frac{1}{nh^2}, \frac{1}{(n\sqrt{h})^{1/2}}\Big)$$

and that is of order $n^{-9/20}$ by (K2). We get

$$\begin{aligned} U_{n,1}(x) &= \sum_{i=1}^{n} K_h(x_i - x)c_n\Delta_n(x_i) \\ &= \frac{1}{(n\sqrt{h})^{1/2}} \sum_{i=1}^{n} K_h(x_i - x)\Delta_n(x_i) \\ \Big(\frac{\sqrt{h}}{n}\Big)^{1/2} U_{n,1}(x) &= \frac{1}{n}\sum_{i=1}^{n} K_h(x_i - x)\Delta_n(x_i) \\ &= \int K_h(u-x)\Delta_n(u)du + O(n^{-9/20}) \\ &= \mathfrak{K}_h\Delta_n(x) + O(n^{-9/20}) \end{aligned}$$

We remark that $\mathfrak{K}_h\Delta_n(x)$ is uniformly bounded by (A3) and (K1). Therefore, we also get

$$\frac{\sqrt{h}}{n}\int U_{n,1}^2(x)dx = \int (\mathfrak{K}_h\Delta_n(x))^2 dx + O(n^{-9/20})$$

b.) As a next step, we investigate $U_{n,2}(x)$. First, we note that the $\dot{\epsilon}_i$ are independent with mean zero and

$$Var(\dot{\epsilon}_i) = Var(\epsilon_i) + Var(\tilde{\epsilon}_i) = 2\sigma^2(x_i)$$

Then we decompose

$$\begin{aligned} U_{n,2}^2(x) &= \Big(\sum_{i=1}^{n} K_h(x_i - x)\dot{\epsilon}_i\Big)^2 \\ &= \sum_{i=1}^{n} K_h^2(x_i - x)\dot{\epsilon}_i^2 + 2\sum_{i<j} K_h(x_i - x)K_h(x_j - x)\dot{\epsilon}_i\dot{\epsilon}_j \\ &= V_{n,2} + W_{n,2} \end{aligned}$$

Therefore, using again Lemma.5.1 and(K2)

$$\begin{aligned}\mathbb{E}\frac{\sqrt{h}}{n}V_{n,2}(x) &= 2\frac{\sqrt{h}}{n}\sum_{i=1}^{n}K_h^2(x_i-x)\sigma^2(x_i)\\ &= 2\sqrt{h}\int_0^1 K_h^2(u-x)\sigma^2(u)du + O\Big(\frac{1}{nh^{3/2}}\Big)\\ &= \frac{2}{\sqrt{h}}\int K^2(y)\sigma^2(x+hy)dy + O\Big(\frac{1}{nh^{3/2}}\Big)\\ &= \frac{2}{\sqrt{h}}\sigma^2(x)\int K^2(y)dy + O(\sqrt{h})\end{aligned}$$

as, by (A2), $|\sigma^2(x+hy)-\sigma^2(x)| \leq h|y|.const = O(h)$, for $y \in [-1,1] = supp(K)$

Remark, that this result is uniform in x. We get

$$\begin{aligned}\mathbb{E}\frac{\sqrt{h}}{n}\int V_{n,2}(x)dx &= \frac{2}{\sqrt{h}}\int\sigma^2(x)dx\int K^2(u)du + O(\sqrt{h})\\ &= B_h + O(\sqrt{h})\end{aligned}$$

Now

$$\begin{aligned}Var\Big(\frac{\sqrt{h}}{n}\int V_{n,2}(x)dx\Big) &= \frac{h}{n^2}\int Var\Big(\sum_{i=1}^{n}K_h^2(x_i-x)\dot{\epsilon}_i^2\Big)dx\\ &= \int\frac{h}{n^2}\sum_{i=1}^{n}K_h^4(x_i-x)Var(\dot{\epsilon}_i^2)dx\\ &\leq \int\frac{1}{nh^3}\frac{1}{n}\sum_{i=1}^{n}K^4\Big(\frac{x_i-x}{h}\Big)dx\max_i(Var(\dot{\epsilon}_i^2))\\ &= \int\frac{1}{nh^3}\int K^4\Big(\frac{u-x}{h}\Big)dudx\max_i(Var(\dot{\epsilon}_i^2)) + O\Big(\frac{1}{n^2h^4}\Big)\\ &= \frac{1}{nh}\int K^4(y)dy\max_i(Var(\dot{\epsilon}_i^2)) + O\Big(\frac{1}{n^2h^4}\Big)\\ &= O\Big(\frac{1}{nh^2}\Big)\end{aligned}$$

We conclude

$$\frac{\sqrt{h}}{n}\int V_{n,2}(x)dx = B_h + O(\sqrt{h}) + O_p(h^{3/2})$$

as $(nh^2)^{-1/2} \sim h^{3/2}$ by (K2).

c.) Now, we consider the term involving $W_{n,2}(x)$, and we prove

$$T_{n,3} = \frac{\sqrt{h}}{n} \int W_{n,2}(x)dx \to \mathcal{N}\Big(0, V\Big) \tag{5.5}$$

(weakly).

First, put

$$W_{ijn} = \begin{cases} \frac{\sqrt{h}}{n} \int_0^1 K_h(X_i - x)K_h(X_j - x)dx\dot{\epsilon}_i\dot{\epsilon}_j, & \text{if } i \neq j; \\ 0, & \text{otherwise} \end{cases}$$

Then

$$T_{n,3} = \sum_{i,j} W_{ijn}$$

According to Theorem 2.1 in de Jong [21] for (5.5) it suffices to prove

$$Var(T_{n,3}) \to V, \tag{5.6}$$

$$\frac{\max_{1\leq i\leq n} \sum_{j=1}^n Var(W_{ijn})}{Var(T_{n,3})} \to 0, \tag{5.7}$$

$$\frac{\mathbb{E}T_{n,3}^4}{(Var(T_{n,3}))^2} \to 3, \tag{5.8}$$

First we prove (5.6), which is a straightforward calculation as follows:

$$\begin{aligned} T_{n,3} &= \frac{\sqrt{h}}{n} \int_0^1 \sum_{i\neq j} K_h(X_i - x)K_h(X_j - x)\dot{\epsilon}_i\dot{\epsilon}_j dx \\ \mathbb{E}T_{n,3} &= \frac{\sqrt{h}}{n}\mathbb{E} \int_0^1 \sum_{i\neq j} K_h(X_i - x)K_h(X_j - x)\dot{\epsilon}_i\dot{\epsilon}_j dx \\ &= 0 \end{aligned}$$

by the assumption of independence $\dot{\epsilon}_i, \dot{\epsilon}_j$ and $\mathbb{E}\dot{\epsilon}_i = 0, \mathbb{E}\dot{\epsilon}_j = 0$.

$$\begin{aligned} T_{n,3}^2 &= \frac{h}{n^2}\Big[\sum_{i\neq j} \dot{\epsilon}_i\dot{\epsilon}_j \int_0^1 K_h(x_i - x)K_h(x_j - x)dx\Big] \\ &= \frac{h}{n^2} \sum_{i\neq j, l\neq k} \dot{\epsilon}_i\dot{\epsilon}_j\dot{\epsilon}_k\dot{\epsilon}_l \int K_h(x_i - x)K_h(x_j - x)dx \int_0^1 K_h(x_k - x)K_h(x_l - x)dx \end{aligned}$$

As $\mathbb{E}\dot{\epsilon}_i\dot{\epsilon}_j\dot{\epsilon}_k\dot{\epsilon}_l = 0$ in the sum except for $i = k, j = l$ or $i = l, j = k$ we get

$$\begin{aligned}
\mathbb{E}T_{n,3}^2 &= \frac{2h}{n^2}\Big[\sum_{i\neq j}\mathbb{E}\dot{\epsilon}_i\dot{\epsilon}_j\int_0^1 K_h(x_i-x)K_h(x_j-x)dx\Big] + 0 \\
&= \frac{8h}{n^2}\Big[\sum_{i\neq j}\sigma^2(x_i)\sigma^2(x_j)\Big(\int_0^1 K_h(x_i-x)K_h(x_j-x)dx\Big)^2\Big] \\
&= 8h\int\int\sigma^2(x)\sigma^2(y)\Big(K_h^{(2)}(x-y)\Big)^2dxdy + O\Big(\frac{n}{h^2}\Big)
\end{aligned}$$

where we have used Lemma.5.1 again and the fact that

$$\begin{aligned}
K_h^{(2)}(x-y) &= \frac{1}{h^2}\int K\Big(\frac{u-x}{h}\Big)K\Big(\frac{u-y}{h}\Big)du \\
&= \begin{cases} 0, & \text{if } |x-y| > 2h; \\ O\Big(\frac{1}{h}\Big), & \text{otherwise} \end{cases}
\end{aligned} \tag{5.9}$$

as K_h has support $[-h, h]$ and integrates to 1. This implies that $\Big(K_h^{(2)}(x-y)\Big)^2$ is Lipschitz with a constant of order $O(h^{-3})$.

For any x, y, we also have by boundedness of σ^2,

$$h\int\sigma^2(x)\Big(K_h^{(2)}(x-y)\Big)^2dx \leq const h\int\Big(K_h^{(2)}(z)\Big)^2dz = O(1)$$

as by (5.9) and Lemma.5.2b)

$$h\int\Big(K_h^{(2)}(z)\Big)^2dz = O(1)\int K_h^{(2)}(z)dz = O(1)$$

mark that by Lemma.5.2a)

$$h\int\Big(K_h^{(2)}(z)\Big)^2dz = K_h^{(4)}(0)$$

Now, using (5.9) and Lipschitz continuity of σ^2 with Lipschitz constant, say L_σ, we get

$$\begin{aligned}\left|h\int \sigma^2(x)\Big(K_h^{(2)}(x-y)\Big)^2dx - h\sigma^2(y)K_h^{(4)}(0)\right| &= \left|h\int_{-2h}^{2h}[\sigma^2(y+z)-\sigma^2(y)]\Big(K_h^{(2)}(z)\Big)^2dz\right.\\ &= 2h^2L_\sigma\int_{-2h}^{2h}\Big(K_h^{(2)}(z)\Big)^2dz\\ &= 2h^2L_\sigma K_h^{(4)}(0)\\ &= O(h)\end{aligned}$$

Finally, straight forward substitution show that $hK_h^{(4)}(0) = K^{(4)}(0)$, such that

$$\mathbb{E}T_{n,3}^2 = 8\int \sigma^4(y)dyK^{(4)}(0) = V$$

Before proving (5.6)-(5.7), we introduce multiplying notation

$$\begin{aligned}L_{ij} &= \sqrt{h}\int_0^1 K_h(x_i-x)K_h(x_j-x)dx\\ &= \begin{cases}\sqrt{h}K^{(2)}(x_i-x_j), & \text{if } i\neq j=1,...,n;\\ 0, & \text{if } i=j\end{cases}\end{aligned}$$

such that we have

$$W_{ijn} = \frac{1}{n}L_{ij}\dot{\epsilon}_i\dot{\epsilon}_j$$

We remark that by (5.9)

$$L_{ij} = \begin{cases}0, & \text{if } |x_i-x_j|>2h;\\ O\Big(\frac{1}{\sqrt{h}}\Big), & \text{otherwise}\end{cases}$$

which we will use frequently below.

First we remark that

$$\begin{aligned}\mathbb{E}T_{n,3}^2 &= \frac{1}{n^2}\sum_{i,j,k,l}L_{ij}L_{kl}\mathbb{E}\dot{\epsilon}_i\dot{\epsilon}_j\dot{\epsilon}_k\dot{\epsilon}_l\\ &= \frac{1}{n^2}\sum_{i,j}L_{ij}^2\mathbb{E}\dot{\epsilon}_i^2\dot{\epsilon}_j^2\\ &= 2\sum_{i,j}\mathbb{E}W_{ij}^2\end{aligned}$$

as $L_{ii} = L_{kk} = 0$ and as, by independence of the $\dot{\epsilon}_i$ and by $\mathbb{E}\dot{\epsilon}_i = 0$, the summands do not vanish for $i = k \neq j = l$ or $i = l \neq j = k$ only.

Now we have to consider

$$\begin{aligned}\mathbb{E}T_{n,3}^4 &= \sum_{i,j,k,l,\mu,\nu,\kappa,\lambda} \mathbb{E}W_{ij}W_{kl}W_{\mu\nu}W_{\kappa\lambda} \\ &= \frac{1}{n^4} \sum_{i,j,k,l,\mu,\nu,\kappa,\lambda} L_{ij}L_{kl}L_{\mu\nu}L_{\kappa\lambda}\mathbb{E}(\dot{\epsilon}_i\dot{\epsilon}_j\dot{\epsilon}_k\dot{\epsilon}_l\dot{\epsilon}_\mu\dot{\epsilon}_\nu\dot{\epsilon}_\kappa\dot{\epsilon}_\lambda)\end{aligned}$$

The terms with $i = j, k = l$,etc vanish by definition of L_{ij}. Also, by independence of the $\dot{\epsilon}_i$, the eight fold expectation vanishes if one index appears only once. So, typical terms which are not vanishing are of the terms form, using $W_{ij} = W_{ji}$,

a) $\mathbb{E}W_{ij}^4, i \neq j$,

b) $\mathbb{E}W_{ij}^2W_{ik}^2, i \neq j \neq k$,

c) $\mathbb{E}W_{ij}^2W_{kl}^2 = W_{ji}^2W_{lk}^2, i \neq j \neq k \neq k$,

d) $\mathbb{E}W_{ij}^2W_{ik}W_{jk}, i \neq j \neq k$,

e) $\mathbb{E}W_{ij}W_{jk}W_{kl}W_{li}, i \neq j \neq k \neq l$

(Compare also to the proof of Proposition 1 of Härdle and Mammer [43])

Under the restriction $i < j, k < l, \mu < \nu, \kappa < \lambda$, we have 1 term of the form a) for each pair $(i, j), i < j$, 3 terms b) corresponding to the choice $i = k = \mu = \kappa$ and $j = l, j = \nu$ or $j = \lambda$ in the eight of the sum above, 3 terms c) corresponding to $(i, j) = (k, l), (i, j) = (\mu, \nu), (i, j) = (\kappa, \lambda)$ respectively. If we relax the restriction then the number of posibilities is multiplied by 4. Let N_d, N_e denote the number of terms of form d), e) corresponding to index triples (i, j, k) and index quadraples (i, j, k, l), respectively

As

$$\begin{aligned}\sum_{ij}\mathbb{E}W_{ij}^4 &= \frac{1}{n^4}\sum_{i,j}L_{ij}^4\mathbb{E}\dot{\epsilon}_i^4\dot{\epsilon}_j^4\\ &= \frac{1}{n^4}\sum_{i,j}L_{ij}^4\mathbb{E}\dot{\epsilon}_i^4\mathbb{E}\dot{\epsilon}_j^4\\ &= O\Big(\frac{1}{h^2}\Big)\frac{1}{n^4}\sum_{i=1}^{4}\sum_{|j-i|\leq n2h}1\\ &= O\Big(\frac{1}{n^2h}\Big)\end{aligned}$$

we have, as in the first sum the term with $i,j\neq k,l$ dominate the whole expression,

$$\mathbb{E}T_{n,3}^4 = 12\sum_{i,j,k,l}\mathbb{E}W_{ij}^2\mathbb{E}W_{kl}^2 + N_d\overset{\neq}{\sum}W_{ij}^2W_{ik}W_{jk} + N_e\overset{\neq}{\sum}W_{ij}W_{jk}W_{kl}W_{li} + O\Big(\frac{1}{n^2h}\Big)$$

where $\sum^{\neq}$ denotes summation over all indices from $1,...,n$ which all assume different values. As

$$\begin{aligned}\mathbb{E}W_{ij}^2W_{ik}W_{jk} &= \frac{1}{n^4}L_{ij}^2L_{ik}L_{jk}\mathbb{E}\dot{\epsilon}_i^3\dot{\epsilon}_j^3\dot{\epsilon}_k^2\\ &= O\Big(\frac{1}{n^4h^2}\Big)\end{aligned}$$

for $|i-k|,|i-j|\leq n2h$ and equal to zero else.

The second sum, running over i,j,k is of order $O\Big(\frac{1}{n}\Big)$. Now, for $i\neq j\neq k\neq l$,

$$\begin{aligned}\mathbb{E}W_{ij}W_{jk}W_{kl}W_{li} &= \frac{1}{n^4}L_{ij}L_{jk}L_{kl}L_{li}\mathbb{E}\dot{\epsilon}_i^2\dot{\epsilon}_j^2\dot{\epsilon}_k^2\dot{\epsilon}_l^2\\ &= \frac{1}{n^4}L_{ij}L_{jk}L_{kl}L_{li}\sigma^2(x_i)\sigma^2(x_j)\sigma^2(x_k)\sigma^2(x_l)\end{aligned}\tag{5.10}$$

with $L_{ij}L_{jk}L_{kl}L_{li} = O\Big(\frac{1}{h^2}\Big)$ for $|i-j|,|j-k|,|k-l|\leq n2h$ and zero else, we get an upper bound for the sum of (5.10) over i,j,k,l of the order

$$O\Big(\frac{1}{n^4h^2}\Big)\sum_{i}\sum_{|i-j|\leq 2nh}\sum_{|j-k|\leq 2nh}\sum_{|k-l|\leq 2nh}1 = O(h)$$

Therefore, the terms of form d), e) are neglijible, and we get

$$\begin{aligned}\mathbb{E}T_{n,3}^4 &= 12\sum_{i,j,k,l}\mathbb{E}W_{ij}^2W_{kl}^2 + O(h)\\ &= 12\sum_{i,j}\Big(\mathbb{E}W_{ij}^2W_{kl}^2\Big)^2 + O(h)\\ &= 3\Big(Var(T_{n,3})\Big)^2 + O(h)\end{aligned}$$

d) Recalling the decomposition of T_n at the beginning of the proof we have

$$T_n = \frac{\sqrt{h}}{n}\int U_{n,1}^2(x)dx + \frac{\sqrt{h}}{n}\int U_{n,2}^2(x)dx + \frac{2\sqrt{h}}{n}\int U_{n,1}(x)U_{n,2}(x)dx$$

We have dealt with the first term which vanishes under the hypothesis H_0, in part a) and with the two components of the second term in part b) and c). We finish the proof by showing that the last term vanishes for $n \to \infty$, and then we can conclude that

$$\begin{aligned}T_n &= \int\Big(\mathfrak{K}_h\Delta_n(x)\Big)^2dx + B_h^0 + T_{n,3} + o_p(1)\\ &= B_h^1 + B_h^0 + T_{n,3} + o_p(1)\end{aligned}$$

and, therefore, T_n is asymptotically $\mathcal{N}(B_h^1 + B_h^0, V)$-distributed.

Let

$$\begin{aligned}T_{n,2} &= \frac{\sqrt{h}}{n}\int U_{n,1}(x)U_{n,2}(x)dx\\ &= \Big(\frac{\sqrt{h}}{n}\Big)^{1/2}\frac{1}{n}\int\sum_{i,j}K_h(x_i-x)K_h(x_j-x)\Delta_n(x_i)\dot{\epsilon}_i dx\\ &= \Big(\frac{\sqrt{h}}{n^3}\Big)^{1/2}\sum_{i,j}K_h^{(2)}(x_i-x_j)\Delta_n(x_i)\dot{\epsilon}_j\end{aligned}$$

We have $\mathbb{E}T_{n,2} = 0$ and, by independence of the $\dot{\epsilon}_j$,

$$\begin{aligned}
\mathbb{E}T_{n,2}^2 &= \frac{\sqrt{h}}{n^3} \sum_{i,j,k,l} K_h^{(2)}(x_i - x_j) K_h^{(2)}(x_k - x_l) \Delta_n(x_i) \Delta_n(x_k) \mathbb{E}\dot{\epsilon}_j \dot{\epsilon}_k \\
&= \frac{\sqrt{h}}{n^3} \sum_{i,j,k} K_h^{(2)}(x_i - x_j) K_h^{(2)}(x_k - x_j) \Delta_n(x_i) \Delta_n(x_k) \sigma^2(x_j) \\
&= \frac{\sqrt{h}}{n^3} \sum_{i=1}^{n} \sum_{|i-j| \le 2nh} \sum_{|k-j| \le 2nh} O\Big(\frac{1}{h^2}\Big) \\
&= O(\sqrt{h})
\end{aligned}$$

Where we have applied (5.9). Therefore $T_{n,2} \to 0$ in probability. □

5.4 The Bootstrap

It is well known that for a moderate sample size the stochastic behaviour of T_n does not work very well and one alternative to the asymptotics is the bootstrap method. Several different bootstrap procedures are possible for this test statistics, however we are going to use the wild bootstrap as it is proposed by Wu [88] (see also Liu [61], Mammen [62], Härdle and Mammen [43]).

First, we estimated the residuals as follow:

$$\begin{aligned}
\hat{\epsilon}_i &= Y_i - \mu^I(x_i) \\
\hat{\tilde{\epsilon}}_i &= Y_i - \mu^{II}(x_i)
\end{aligned}$$

centering the residual by their sample mean, we achieve

$$\hat{\epsilon}_i^0 = \hat{\epsilon}_i - \frac{1}{n} \sum_{j=1}^{n} \hat{\epsilon}_j \tag{5.11a}$$

$$\hat{\tilde{\epsilon}}_i^0 = \hat{\tilde{\epsilon}}_i - \frac{1}{n} \sum_{j=1}^{n} \hat{\tilde{\epsilon}}_j \tag{5.11b}$$

Then, we construct our bootstrap samples

$$Y_i^* = \mu_g^I(x_i) + \hat{\epsilon}_i^{*0}$$
$$\tilde{Y}_i^* = \mu_g^{II}(x_i) + \hat{\tilde{\epsilon}}_i^{*0}$$

while, following Franke [29], we use the oversmooth estimation μ_g^I, μ_g^{II}, where g is chosen such that $h, g \to 0, \frac{h}{g} \to 0$ for $n \to \infty$. In practice we choose $g > h$, e.g. $(g = 2h)$

For the construction of $\hat{\epsilon}_i^{*0}, \hat{\tilde{\epsilon}}_i^{*0}$, we define an arbitrary distribution $\hat{F}_i$, such that,

$$\mathbb{E}_{\hat{F}_i} Z = 0,$$
$$\mathbb{E}_{\hat{F}_i} Z^2 = (\hat{\epsilon}_i^0)^2,$$
$$\mathbb{E}_{\hat{F}_i} Z^3 = (\hat{\epsilon}_i^0)^3.$$

and analogously with $\hat{\tilde{\epsilon}}_i^0$ respectively. We use two-point distribution which is uniquely determined by these requirements. The two-point distribution $\hat{F}_i$

$$\hat{F}_i = \gamma\delta_a + (1-\gamma)\delta_b$$

is defined through three parameters a, b, γ, and where δ_a, δ_b denote point measures at a, b, respectively. Some algebra reveals that the parameters a, b, γ at each location x_i are given by

$$a = \hat{\epsilon}_i^0(1-\sqrt{5})/2,$$
$$b = \hat{\epsilon}_i^0(1+\sqrt{5})/2,$$
$$\gamma = (5+\sqrt{5})/10$$

These parameters ensure that $\mathbb{E}\hat{\epsilon}_i^{*0} = 0, \mathbb{E}(\hat{\epsilon}_i^{*0})^2 = (\hat{\epsilon}_i^0)^2$ and $\mathbb{E}(\epsilon^{*0})^3 = (\hat{\epsilon}_i^0)^3$. Similarly for the construction of $\hat{\tilde{\epsilon}}_i^{*0}$.

Now, the bootstrap test statistics can be constructed as follows. From (5.3) we derive, using

$$T_n = nh^{1/2} \int \big(\mu_h^I(x) - \mu_h^{II}(x)\big)^2 dx \tag{5.12a}$$

$$\sim h^{1/2} \sum_{i=1}^{n} \big(\mu_h^I(x_i) - \mu_h^{II}(x_i)\big)^2 \tag{5.12b}$$

$$= \frac{1}{nh^{1/2}} \sum_{i=1}^{n} \big(\sqrt{nh}\big(\mu_h^I(x_i) - \mu_h^{II}(x_i)\big)^2 \tag{5.12c}$$

$$= \frac{1}{nh^{1/2}} \sum_{i=1}^{n} \big(\sqrt{nh}\big(\mu_h^I(x_i) - m(x_i) + m(x_i) - \mu_h^{II}(x_i)\big)^2 \tag{5.12d}$$

$$\sim h^{1/2} \sum_{i=1}^{n} \big(\mu_h^I(x_i) - \mu_g^I(x_i) + \mu_g^{II}(x_i) - \mu_h^{II}(x_i)\big)^2 \tag{5.12e}$$

under the hypothesis H_0. We use two forms of the test statistics based on (5.12b) or (5.12e) with the bootstrap samples. For now on we call them as $T1_n$ and $T2_n$ respectively, and we set

$$t1_n^* = h^{1/2} \sum_{i=1}^{n} \big(\mu_h^{*I}(x_i) - \mu_h^{*II}(x_i)\big)^2 \tag{5.13}$$

$$t2_n^* = h^{1/2} \sum_{i=1}^{n} \big(\mu_h^{*I}(x_i) - \mu_g^I(x_i) + \mu_g^{II}(x_i) - \mu_h^{*II}(x_i)\big)^2 \tag{5.14}$$

From the Monte Carlo approximation of $\mathcal{L}^*(t1_n^*)$ we construct a $(1-\alpha)$ quantile $\hat{t}1_\alpha$ and reject the null hypothesis if $T1_n > \hat{t}1_\alpha$. Similarly for $T2_n > \hat{t}2_\alpha$.

Proposition 5.2. *Assume (A1)-(A2), (K1)-(K2), then*

$$d_k\big(\mathcal{L}(\sqrt{nh}(\mu_h^I(x_0) - m^I(x_0))), \mathcal{L}(\sqrt{nh}(\mu_h^{*I}(x_0) - \mu_g^I(x_0))|Y_1, ..., Y_n)\big) \to 0 \tag{5.15}$$

Similarly for μ_h^{II}, m_h^{II}

This result is well known. Essentially it goes back to Härdle and Bowman [42] which showed that the bootstrap does not reproduce a correct approximation of bias if g is of the same optimal order as $h \sim n^{-1/5}$. Franke and Härdle [28] showed in the analogous spectral estimation problem that the bias problem dissapears if $h/g \to 0$ for $n \to \infty$.

We also have anologously to Theorem 2 of Härdle and Mammen [43] the following result which shows the consistency of the bootstraping approximation of our test statistic T_n. The proof goes along the same line of arguments as in Proposition 5.1. In particular, additional assumption (A4), implies $sup_i\epsilon_i^2, sup_i\tilde{\epsilon}_i^2 = O_p(log\ \ n)$ and $\mathbb{E}\epsilon_i^8, \mathbb{E}\tilde{\epsilon}_i^8 \leq const < \infty$ for all $i = 1, 2, ...$ Then, we can use again theorem of de Jong.

(A4) $\mathbb{E}\exp(t\epsilon_i), \mathbb{E}\exp(t\tilde{\epsilon}_i)$ are uniformly bounded in $i = 1, 2, ...$ for all $|t|$ small enough.

Proposition 5.3. *Assume (A1)-(A4), (K1),(K2). Then*

$$d(\mathcal{L}(T_n^*|Y_1, ..., Y_n), \mathcal{N}(B_h, V)) \rightarrow_p 0$$

where B_n, V are as in the Proposition 5.1

5.5 The two - dimensional case

We want to apply the test developped in Section 5.1 to the two-dimensional case. That is a straight forward exercise involving only more cumbersome notation. Therefore, we only give the statement but do not give the proof. Compare also Härdle and Mammen [43] who considered also the d-dimensional case in their testing problem. First let us state the assumptions. (A1), (A2) and (K1) are not changed at all except for m^I, m^{II}, σ^2 and K being functions of $x \in \mathbb{R}^2$. (A3) is replaced by

(A3') m^{II} can be written as $m^{II}(x) = m^I(x) + c_n\Delta_n(x)$ with $c_n = (nh)^{1/2}$ and $\Delta_n(x)$ bounded uniformly in x and n

and (K2) is replaced by
(K2') The bandwidth h fulfills $h = h_n \sim cn^{-1/6}$

Remark that (K2') specifies the rate of mse-optimal bandwidth in two dimensions. Then, our test statistic is now, with $x = (x_1, x_2)^T$,

$$T_n = nh \int\int (\mu^I(x) - \mu^{II}(x))^2 dx_1 dx_2$$

where μ^I, μ^{II} denote the Priestley-Chao estimate based on Y_{ij}^I, Y_{ij}^{II} respectively, i.e., of the form

$$\mu_h(x) = \frac{1}{n^2} \sum_{i,j=1}^{n} K_h(x - x_{ij}) Y_{ij}$$

with $K_h(u) = h^{-2}K(u/h), h \in \mathbb{R}$ and $x_{ij} = \left(\frac{i}{n}, \frac{j}{n}\right), i, j = 1, ..., n$

Theorem 5.1. *Assume (A1),(A2),(A3'), (K1),(K2'). Then,*

$$d(\mathcal{L}(T_n), \mathcal{N}(B_h, V)) \to 0$$

with $B_h = B_h^0 + B_h^1$,

$$B_h^0 = \frac{2}{h} \int\int \sigma^2(x) dx_1 dx_2 - \int\int K^2(u) du_1 du_2$$
$$B_h^1 = \int\int \left(\mathfrak{K}_h \Delta_n(x)\right)^2 dx_1 dx_2$$
$$V = 8K^{(4)}(0) \int\int \sigma^4(x) dx_1 dx_2$$

Analogously, the bootstrap result of Section 5.4 would be also generalized to two-dimensions.

5.6 Simulation

Consider a large image $Y = \{Y(i,j), 1 \le i \le M, 1 \le j \le N\}$. To detect a defect, we split it into two parts-roughly one half each - horizontally first and later on, also vertically. Let $\tilde{M}$ be the split coordinate for the horizontal split, i.e., we consider two subimages

$$Y^I(i,j) = Y(i,j), i = 1, ..., \tilde{M}, j = 1, ..., N$$
$$Y^{II}(i,j) = Y(i + \tilde{M}, j), i = 1, ..., M - \tilde{M}, j = 1, ..., N$$

Later on, we consider an analogous split vertically at $j = \tilde{N}$. We discuss the choice of $\tilde{M}, \tilde{N}$ below.

As short-hand notation, we use $Y_{\cdot j}, Y^I_{\cdot j}, Y^{II}_{\cdot j}$ respectively for the j-th column of Y, Y^I, Y^{II}. Mark that for a horizontal split we have $Y_{\cdot j} = (Y^I_{\cdot j}, Y^{II}_{\cdot j})^T$. Now, we model for any $j = 1, ..., N$ $Y^I_{\cdot j}$ and $Y^{II}_{\cdot j}$ as in (5.1). We test for each sample $Y^I_{\cdot j}, Y^{II}_{\cdot j}$ if $H_0 : m^I = m^{II}$ holds or not. If the hypothesis is rejected, we mark the column index j as j^d.

Similarly, if we start with a vertical split, we have

$$Y^I(i,j) = Y(i,j), i = 1,...,M, j = 1,...,\tilde{N}$$
$$Y^{II}(i,j) = Y(i,j+\tilde{N}), i = 1,...,M, j = 1,...,N-\tilde{N}$$

Then we compare the two parts of rows $Y^I_{.j}, Y^{II}_{.j}$, and we mark i as i^d if $H_0 : m^I = m^{II}$ is rejected. Let $i^d_{min} = \min i^d, i^d_{max} = \max i^d, j^d_{min} = \min j^d, j^d_{max} = \max j^d$ then the set of $\{(i^d_{min}, j^d_{min}), ..., (i^d_{max}, j^d_{max})\}$ is the defect area of the image.

The problem left to be explained in this procedure is finding the $\tilde{M}$ and $\tilde{N}$. Due to the illumination, when we observe the image line based, we are going to have a linear trend in the series (see Figure 5.1 as an illustration). We have to remove this trend, line by line at the pre-processing step, such that misleading defect detection due to illumination can be avoided.

We can not always divide the image exactly into two in the middle, especially for the regular and semi regular image texture. Since, when we fold the series, i.e., $Y_1, ..., Y_{\tilde{N}}$ to the $Y_{\tilde{N}+1}, ..., Y_{2\tilde{N}}$ and compare them directly under the null hypothesis, the rejection of the null hypothesis can be misleading due to a wrong shift in the series (see Figure 5.2). To avoid this, we are going to use the periodicity in the image. Let T_M, T_N be periods in the image horizontally respectively vertically (compare to the section 4.1)

For example, we consider the vertical division. We multiply TN by $k \in \mathbb{Z}$ such that $kTN \leq \lfloor N/2 \rfloor$, let this number be N_k. For the moment we have

$$Y^I(i,j) = Y(i,j), i = 1,...,M, j = 1,...,N_k$$
$$Y^{II}(i,j) = Y(i,j+N_k), i = 1,...,M, j = 1,...,N-N_k$$

Let $E_{N_k} = Y^I(i,j) - Y^{II}(i,j)$.

Now, we reduce the N_k by l, where $l = 1, ..., T_N - 1$, such that the mean square error of $E_{N_k - l}$ is minimum. The length, which $N_k - l$ gives the minimum mean square error for E_{N_k-l}, is the $\tilde{N}$. We use a similar procedure for finding the $\tilde{M}$.

The first figure in Figure.5.1 shows us the problem of illumination. This can be seen through the existance of linear trend in the series, which represents a line in the image.

Since our procedure is a line based checking, this series certaintly will be catagorized as a defect. Furthermore, every time we compare two images with different sources of light, the hypothesis will be rejected and the whole image will be catagorized as defect. But this is not as we intend it to be. Removing this trend, see the second figure in Figure 5.2 will secure us from this misleading error detection.

We have another misleading error, namely the shifting error. This error arises because we only want to use one image (without reference image) to detect an error. As we explain above, when we split the image into two parts, the shifting error happend when we were not careful enough to decide the point where the series should be *folded* into two. The first figure in Figure.5.2 gives us an illustration if the series is simply folded in the middle. Then if we use these two series in hypothesis test, the hypothesis can be rejected but there is no defect in the image. Compare to the second figure in Figure.5.2. We shift the series such that the mean of the square difference between two series is minumum, testing these two series gives a more resonable result than those previous two series.

Algorithm 5.1. *:* **Defect Detection**
for *j = 1:N*

1. *Remove linear trend in* $Y_{\cdot j}$
2. *Find* $\tilde{M}$
3. *Set up* $Y^I_{\cdot j}$ *and* $Y^{II}_{\cdot j}$
4. *Set* $x_i = 1/N_1, 2/N_1, ..., 1$ *where* $N_1 = \min(\tilde{N}, N - \tilde{N})$ *is the length of the series* $Y^I_{\cdot j}$ *and* $Y^{II}_{\cdot j}$*, where we drop part of the larger of the two sequences* $Y^I_{\cdot j}, Y^{II}_{\cdot j}$ *to get them of equal length.*
5. *Compute the* $\mu^I_h(x), \mu^{II}_h(x)$ *using (5.2a) and (5.2b) respectively.*
6. *Use the same equations calculate the* $\mu^I_g(x), \mu^{II}_g(x)$ *with* $g = 2h$
7. *Compute the centering error using (5.11a) - (5.11b)*

   for *B = 1:500*

   8a. Construct wild bootstrap data $Y^{*I}_{\cdot j}$ *and* $Y^{*II}_{\cdot j}$

8b. Compute $\mu_h^{*I}(x), \mu_h^{*II}(x)$

8c. Compute test statistics, $t1^*$ *using (5.13) or* $t2^*$ *using (5.14) for the bootstrap data*

end

9. Construct the $(1-\alpha)$*-quantile* $\hat{t}1_\alpha$ *or* $\hat{t}2_\alpha$

10. Construct the $T1_n$ *using (5.12b) or* $T2_n$ *using (5.12e)*

11. Reject the hypothesis if $T1_n > \hat{t}1_\alpha$ *or* $T2_n > \hat{t}2_\alpha$

12. Mark the index j*.*

13. Cluster the defect set to avoid the outliers in this set, we define jcluster as the set without outliers.

end

Find $jcluster^d_{min}, jcluster^d_{max}$ *from the clustering set*
Use the similar procedure to get the defect set columwise and find $icluster^d_{min}, icluster^d_{max}$
The set $\{(icluster^d_{min}, jcluster^d_{min}), ..., (icluster^d_{max}, jcluster^d_{max})\}$ *is the defect area of the image.*

We give five different examples of error detections on a texture in Figure 5.3. Figure 5.3a, represents noisy regular image, Figures.5.3 b and c, represent semi regular texture. We can see in both images that our procedure is invariant to the lighting (b) and rotation (c). The last two figures represent the error detection for random texture.

However, for this moment, this method still has a drawback. Particularly for detecting a defect for natural texture, such as wood or if the contrast of the defect area to the background is too feeble, see Figure5.4).

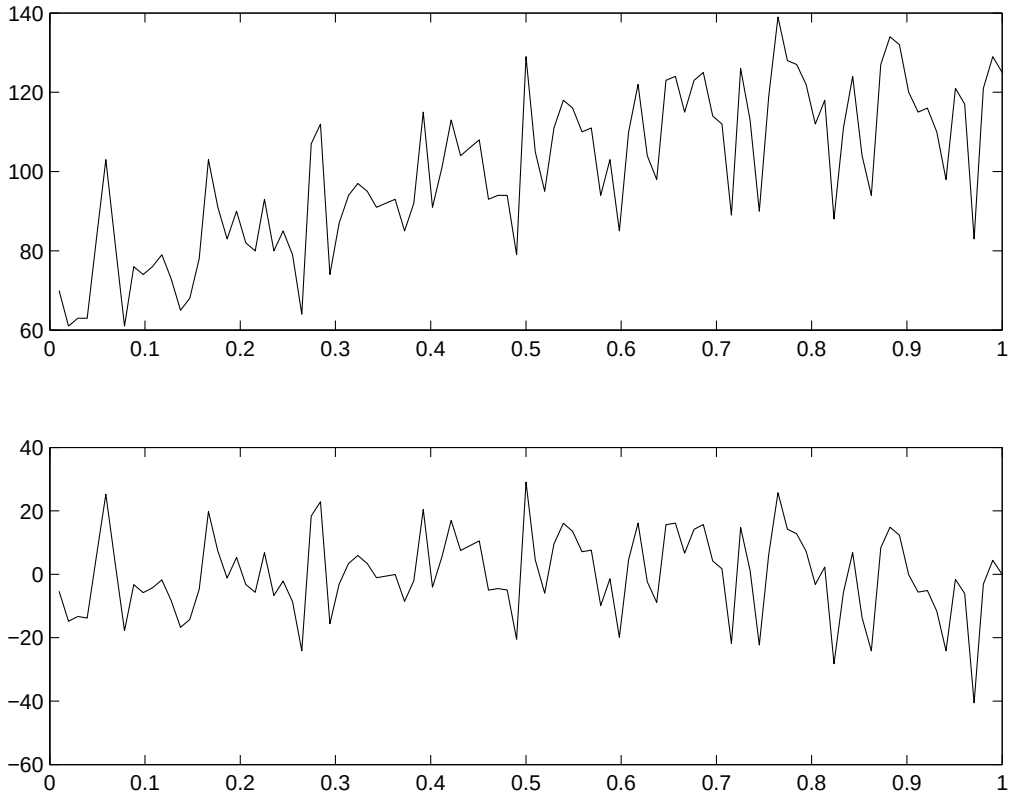


Figure 5.1: The series with linear trend, and after the trend is removed

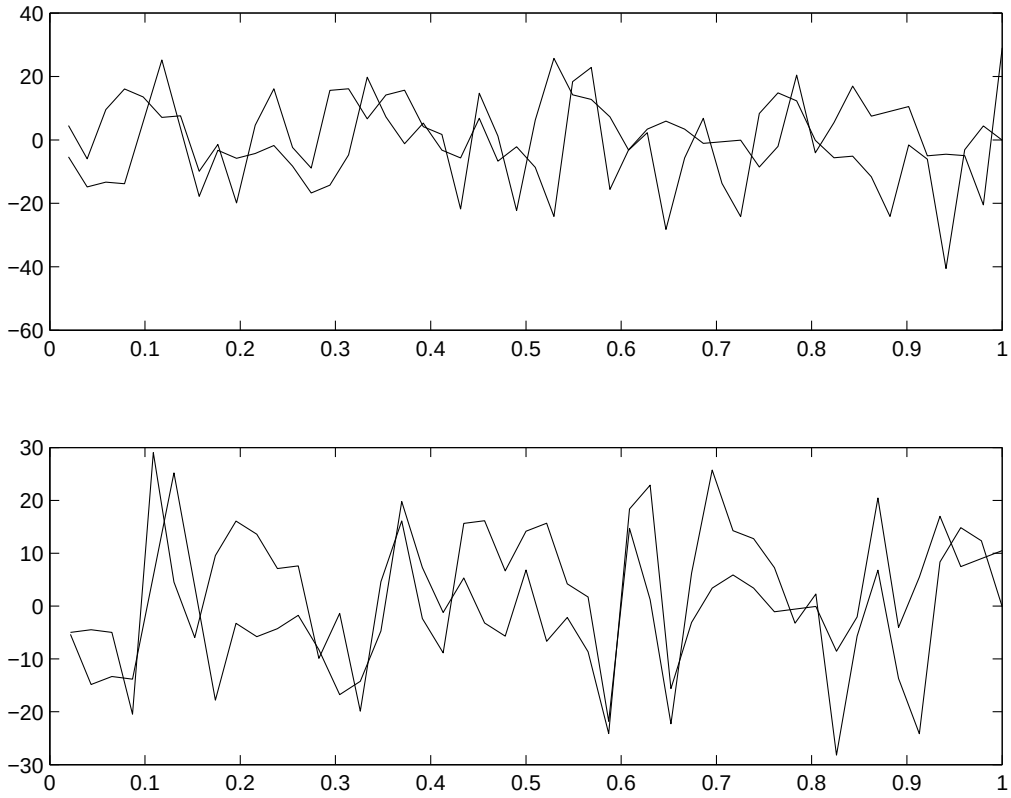


Figure 5.2: The devision of series in the middle and based on minimum mean square error

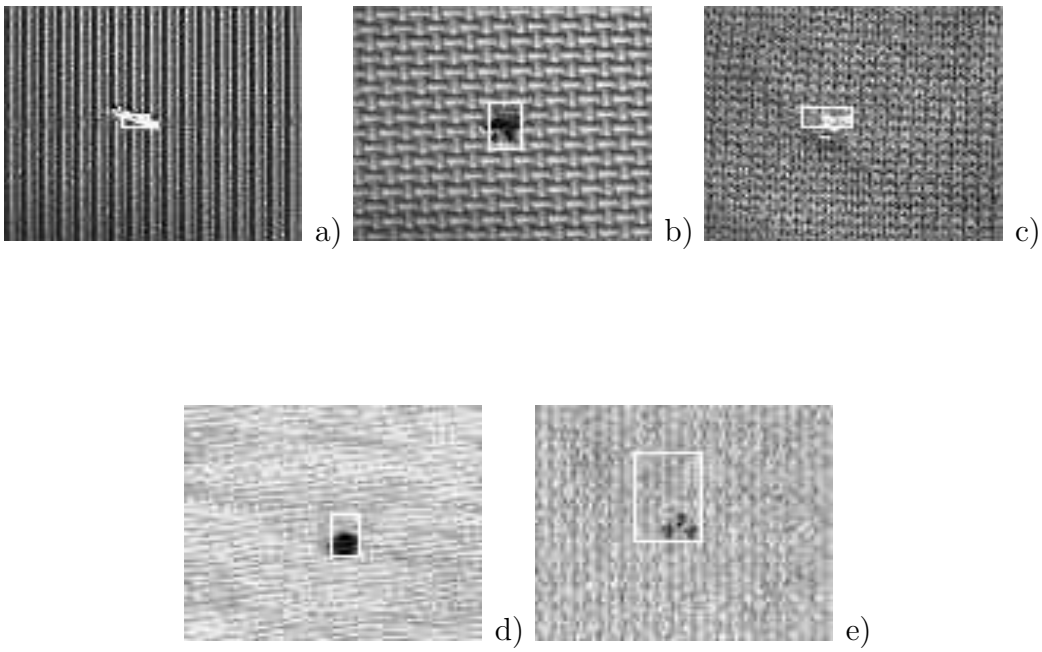


Figure 5.3: Some examples of defect detection

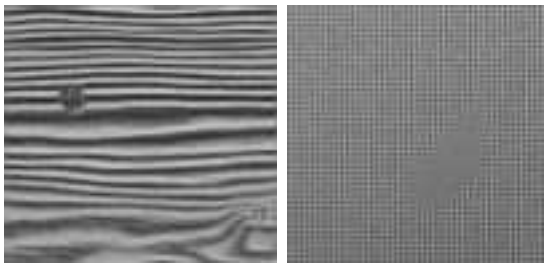


Figure 5.4: Some examples of undetected errors

Appendix A

Tables

Table A.1: Summary of Denoising of DFG SPP 1114's Images Test Simple Methods as an Comparation

Images	σ	Method	Mask	L_1	L_2	L_∞	LDP	PSNR
		ND		0.135830	0.041172	0.994760	0.218550	13.85410
		MF	3x3	0.068648	0.013247	1.000000	0.053055	18.79570
	0.25	AF	3x3	0.113340	0.022129	0.672370	0.092804	16.51880
		GF	5x5	0.117560	0.022950	0.682280	0.109680	16.36010
		MWH		0.092641	0.012659	0.739460	0.026443	18.94920
		ND		0.240880	0.120410	1.000000	0.425400	9.193500
Chess Board		MF	5x5	0.112820	0.029973	1.000000	0.130390	15.24810
&	0.50	AF	3x3	0.178050	0.046426	0.814000	0.244290	13.34340
Circle		GF	5x5	0.203740	0.066621	0.897090	0.330230	11.77010
		MWH		0.158820	0.034721	0.828540	0.123030	14.60930
		ND		0.348370	0.230340	1.000000	0.553220	6.376500
		MF	7x7	0.165460	0.059280	1.000000	0.256180	12.28570
	1.00	AF	5x5	0.242440	0.084016	0.775890	0.521330	10.72100
		GF	5x5	0.294640	0.132950	0.982450	0.488450	8.734000
		MWH		0.229340	0.075251	0.857430	0.532810	11.22260

Table A.2: Summary of Denoising of DFG SPP 1114's Images Test Simple Methods as an Comparation-continue

Images	σ	Method	Mask	L_1	L_2	L_∞	LDP	PSNR
		ND		0.175300	0.049107	0.933170	0.275430	13.08860
		MF	5x5	0.068476	0.009409	0.804500	0.027679	20.22280
	0.25	AF	5x5	0.070797	0.009493	0.649910	0.028763	20.25880
		GF	5x5	0.115520	0.021672	0.646030	0.092346	16.65940
		MWC		0.062235	0.007342	0.631150	0.016785	21.36900
		ND		0.286120	0.123060	0.976000	0.535900	9.098800
		MF	9x9	0.092507	0.019059	0.976470	0.072800	17.14780
Camera	0.5	AF	7x7	0.106900	0.018736	0.649760	0.070541	17.29900
		GF	5x5	0.190630	0.056378	0.818570	0.303790	12.50220
		MWC		0.098895	0.016254	0.584560	0.054047	17.91210
		ND		0.380970	0.202240	0.986230	0.697180	6.941400
		MF	9x9	0.139290	0.036007	0.976470	0.150650	14.35770
	1.0	AF	9x9	0.149140	0.034058	0.666100	0.221270	14.67530
		GF	5x5	0.190630	0.056378	0.818570	0.303790	12.50220
		MWC		0.142490	0.031699	0.618950	0.212140	15.00570
		ND		0.173490	0.047236	0.927370	0.245250	13.25740
		MF	7x7	0.078725	0.012743	0.996080	0.047607	18.93170
	0.25	AF	3x3	0.080461	0.010796	0.622370	0.023087	19.68070
		GF	5x5	0.114660	0.021038	0.816210	0.085831	16.77320
		MWC5		0.072327	0.009231	0.550690	0.022278	20.36630
		ND		0.281770	0.121320	1.000000	0.477290	9.160600
		MF	7x7	0.108160	0.021744	1.000000	0.081955	16.73760
Satellite	0.50	AF	5x5	0.118780	0.021304	0.781670	0.075684	16.77610
		GF	5x5	0.186630	0.056236	0.900500	0.286760	12.49390
		MWC4		0.115810	0.019660	0.600810	0.063370	17.14110
		ND		0.377950	0.204640	1.000000	0.621020	6.890100
		MF	9x9	0.146840	0.038294	1.000000	0.163760	14.17970
	1.00	AF	7x7	0.167190	0.038670	0.744850	0.212490	14.13480
		GF	7x7	0.254040	0.101540	0.972160	0.433070	9.944100
		MWC4		0.166060	0.036942	0.636620	0.196590	14.33230

Table A.3: Summary of Denoising of DFG SPP 1114's Images Test Simple Methods as an Comparation-continue

Images	σ	Method	Mask	L_1	L_2	L_∞	LDP	PSNR
		ND		0.107170	0.030582	0.843760	0.172920	15.14550
		MF	9x9	0.023404	0.002688	0.938080	0.010727	25.72340
	0.25	AF	5x5	0.025990	0.002900	0.606020	0.009125	25.32250
		GF	5x5	0.036631	0.003686	0.333390	0.000854	24.38220
		MWC5		0.034332	0.002980	0.486820	0.003952	25.26810
		ND		0.173020	0.071799	0.999300	0.336380	11.43890
		MF	9x9	0.040887	0.006806	0.938080	0.021744	21.64130
Radon	0.5	AF	7x7	0.040196	0.005139	0.623540	0.013992	22.78420
		GF	5x5	0.110850	0.030543	0.749180	0.180630	15.13460
		MWC3		0.046056	0.005568	0.576310	0.015274	22.51010
		ND		0.224370	0.109530	0.999990	0.437750	9.604700
		MF	9x9	0.072371	0.017102	0.987500	0.061234	17.47790
	1.0	AF	7x7	0.055703	0.008770	0.666210	0.028412	20.32730
		GF	5x5	0.142120	0.047499	0.865270	0.270290	13.15700
		MWC3		0.060033	0.008737	0.706360	0.028748	20.32900
		ND		0.181400	0.050086	0.877190	0.265380	13.00290
		MF	5x5	0.077767	0.010763	0.792160	0.027725	19.67190
	0.25	AF	5x5	0.072506	0.009258	0.553980	0.019714	20.31900
		GF	5x5	0.118630	0.022172	0.629840	0.095657	16.53710
		MWC5		0.068344	0.008211	0.458630	0.014618	20.85380
		ND		0.292610	0.123200	0.938580	0.516190	9.093800
		MF	9x9	0.099539	0.018846	0.866670	0.063980	17.23630
	0.50	AF	7x7	0.103770	0.016324	0.588200	0.042206	17.85510
		GF	5x5	0.189030	0.055530	0.798950	0.299680	12.54950
		MWC5		0.100750	0.015135	0.525410	0.033432	18.19460
		ND		0.385590	0.198140	0.950750	0.671440	7.030200
		MF	9x9	0.143270	0.034954	0.933330	0.154500	14.51860
	1.00	AF	9x9	0.139520	0.027755	0.643450	0.124880	15.52590
		GF	5x5	0.249090	0.093107	0.861590	0.448000	10.28620
		MWC3		0.138120	0.026921	0.546940	0.119860	15.67910

Table A.4: Summary of Denoising of DFG SPP 1114's Images Test Using Kernel Based Methods

Images	σ	Method	L_1	L_2	L_∞	LDP	PSNR
Chess Board & Circle	0.25	CV2P	0.0226	0.0021	0.3537	0.0166	22.3414
		AC2P	0.0245	0.0025	0.3588	0.0250	21.5323
		AC3P	0.0282	0.0035	0.3747	0.0742	20.1557
	0.50	CV2P	0.0308	0.0027	0.3754	0.0301	21.2213
		AC2P	0.0317	0.0033	0.3841	0.0469	20.4257
		AC3P	0.0351	0.0038	0.3780	0.0666	19.7913
	1.00	CV2P	0.0591	0.0088	0.7201	0.1253	16.1309
		AC2P	0.0484	0.0059	0.4282	0.0861	17.8785
		AC3P	0.0521	0.0069	0.5556	0.0923	17.1664
Camera	0.25	CV2P	0.0190	0.0011	0.3128	0.0054	25.1760
		AC2P	0.0178	0.0009	0.2786	0.0036	25.9924
		AC3P	0.0185	0.0009	0.2952	0.0032	26.0955
	0.50	CV2P	0.0271	0.0021	0.3846	0.0203	22.2690
		AC2P	0.0249	0.0014	0.3394	0.0067	24.1572
		AC3P	0.0255	0.0014	0.3408	0.0067	24.0464
	1.00	CV2P	0.0487	0.0072	0.6332	0.0824	16.9732
		AC2P	0.0354	0.0026	0.4054	0.0191	21.4240
		AC3P	0.0363	0.0027	0.4122	0.0215	21.1998
Satellite	0.25	CV2P	0.0261	0.0016	0.3442	0.0076	23.5964
		AC2P	0.0284	0.0018	0.3412	0.0095	22.9051
		AC3P	0.0280	0.0018	0.3378	0.0079	23.1041
	0.50	CV2P	0.0362	0.0029	0.3816	0.0262	20.8750
		AC2P	0.0346	0.0026	0.3722	0.0198	21.3672
		AC3P	0.0348	0.0026	0.3525	0.0194	21.3713
	1.00	CV2P	0.0575	0.0079	0.6919	0.0906	16.5652
		AC2P	0.0428	0.0037	0.4499	0.0324	19.9285
		AC3P	0.0446	0.0040	0.4465	0.0381	19.5517

Table A.5: Summary of Denoising of DFG SPP 1114's Images Test Using Kernel Based Methods - continue

Images	σ	Method	L_1	L_2	L_∞	LDP	PSNR
		CV2P	0.0090	0.0002	0.1873	0.0002	31.7020
	0.25	AC2P	0.0074	0.0002	0.1496	0.0000	32.5745
		AC3P	0.0086	0.0003	0.2062	0.0004	31.0646
		CV2P	0.0175	0.0012	0.3754	0.0132	24.7076
Radon	0.50	AC2P	0.0112	0.0004	0.1896	0.0007	29.7841
		AC3P	0.0127	0.0006	0.2707	0.0031	28.1169
		CV2P	0.0386	0.0057	0.7419	0.0714	17.9901
	1.00	AC2P	0.0252	0.0022	0.6889	0.0256	22.0530
		AC3P	0.0280	0.0026	0.6889	0.0325	21.3616
		CV2P	0.0296	0.0020	0.2737	0.0086	22.6206
	0.25	AC2P	0.0292	0.0019	0.2493	0.0073	22.7890
		AC3P	0.0301	0.0020	0.2883	0.0074	22.6406
		CV2P	0.0369	0.0028	0.3702	0.0194	21.0232
Barbara	0.50	AC2P	0.0344	0.0024	0.2785	0.0112	21.7952
		AC3P	0.0350	0.0024	0.2731	0.0112	21.7795
		CV2P	0.0534	0.0067	0.6658	0.0627	17.2933
	1.00	AC2P	0.0423	0.0034	0.5743	0.0204	20.2962
		AC3P	0.0442	0.0036	0.5743	0.0229	19.9829

Appendix B

Algorithms

This algorithms are taken from [74]

B.1 Gibbs Sampler

Step 1: choose an initial image x and define a visiting scheme $\{s_n\}_{n\geq 1}$
Step 2: repeat
for i = 1 to m do
Step 2.1: for all labels $l = 0, ..., L-1$
compute $p_l = \Pi(X_{s_i} = l | X_r = x_r, r \in \mathcal{N}(s_i)$
Step 2.2: set x_{s_i} to l with probability p_l
until x satisfies the stability criterion

B.2 Metropolis Sampler

Step 1: choose an initial image x and define a visiting scheme $\{s_n\}_{n\geq 1}$
Step 2: repeat
for i = 1 to m do
Step 2.1: chooce $l \in \mathcal{L} \backslash x_{s_i}$ uniformly at random
and set $\tilde{x}_{s_i} = l, \tilde{x}_s = x_s$ for all $s \neq s_i$
Step 2.2: calculate $p := \min\{1, \frac{\Pi(xt)}{\Pi(x)}\}$
Step 2.3: set x_{s_i} to l with probability p
until x satisfies the stability criterion

B.3 Exchange Algorithm

Step 1: choose an initial image x and define a visiting scheme $\{s_n\}_{n\geq 1}$
Step 2: repeat
for i = 1 to m do
Step 2.1: select two sites such that uniformly at random
from the set of all sites s with $x_s \neq x_t$
Step 2.2: Set $\tilde{x}_u := x_u$ for all $u \neq s, t$ and $\tilde{x}_s := x_t$ if $\tilde{x}_t := x_s$
Step 2.3: compute $p := \min 1, \frac{\Pi(xt)}{\Pi(x)}\}$
Step 2.4: Set x to $\tilde{x}$ with probability p
until x satisfies the stability criterion

Bibliography

[1] M. Abramowitz and I.A.Stegun. *Handbooks of Mathematical Functions, with Formulas, Graphs, and Mathematical Tables*. Dover, New York, U.S.A., 1970.

[2] C. O. Acuna. Texture modeling using gibbs distributions. *CVGIP : Graphical Models and Image Processing*, 54,3:210–222, 1992.

[3] T. W. Anderson. *The Statistical Analysis of Time Series*. Wiley, New York, U.S.A., 1971.

[4] L. Anselin. *Spatial Econometrics: Methods and Models*. Kluwer Academic Publisher, Dordrecht, 1981.

[5] C. Bahlmann, G. Heidemann, and H. Ritter. Artificial neural networks for automated quality control of textile seams. *Pattern Recognation Letters*, 32:1049–1060, 1999.

[6] J. E. Besag. Spatial interaction and the statistical analysis of lattice system. *Journal of The Royal Statistical Society, series B*, 36:192–326, 1974.

[7] J. E. Besag. On the statistical analysis of dirty pictures. *Journal of The Royal Statistical Society, series B*, 48:259–302, 1986.

[8] P. Brodatz. *Textures: A photographic album for artists and designers*. Dover, New York, U.S.A., 1966.

[9] E. Candès. *Ridgelets: theory and applications*. PhD thesis, Department of Statistics, Standford University, 1998. Website. `http://www.acm.caltech.edu/~emmanuel/publications.html`.

[10] E. Candès. Harmonic analysis of neural networks. *Applied and Computational Harmonic Analysis*, 6:197–218, 1999.

[11] E. Candès. Monoscale redgelets for the representation of images with edges. Website, 1999. `http://www.acm.caltech.edu/~emmanuel/publications.html`.

[12] R. Chellappa and R. L. Kashyap. Texture synthesis using spatial interaction models. *IEEE Transactions on on Acoust., Speech and Signal Proc.*, ASSP-33:194–203, Feb. 1985.

[13] D. Chetverikov. Structural defects: General approach and application to textile inspection. Website, 2000. `http://visual.ipan.sztaki.hu/publ/publ.html`.

[14] D. Chetverikov and A. Hanbury. Finding defects in texture using regularity and local orientation. Website, 2002. `http://visual.ipan.sztaki.hu/publ/publ.html`.

[15] S. T. Chiu. Detecting periodic components in a white gaussian time series. *Journal of Royal Statistics Society series B*, 51,2:249–259, 1989.

[16] C. Chu, I. Glad, F. Godtliebsen, and J. Marron. Edge-preserving smoothers for image processing. *Journal of The American Statistical Association*, 93,442:526–556, 1998.

[17] C. Chu and J. Marron. Choosing a kernel regression estimator. *Statistical Science*, 6:404–436, 1991.

[18] G. C. Cross and A. K. Jain. Markov random field texture models. *IEEE Transactions Pattern Analysis Machine Intelligence*, 5:25–39, 1983.

[19] C. Daul, R. Rösch, B. Claus, J. Grotepaβ, U. Knaak, and R. Föhr. A fast image processing algorithm for quality control of woven textiles. In e. P. Levi, editor, *Mustererkennung, DAGM*. Springer Verlag, 1998.

[20] J. S. DeBonet. Multiresolution sampling procedure for analysis and synthesis of texture images. In *Computer Graphics*, pages 361–368. ACM SIGGRAPH, 1997. Website. `http://www.debonet.com/research/publications/1997/DeBonet-SIGGRAPH97-TextureSynthesis.pdf`.

[21] P. DeJong. A central limit theorem for generalized quadratic forms. *Probability Theory Related Fields*, 75:383–400, 1987.

[22] M. Delgado. Testing the equality of nonparametric regression curves. *Statistics & Probabiltiy Letters*, 17:199–204, June 1993.

[23] B. Efron and R. J. Tibshirani. *An Introduction to the Bootstrap*. Chapman & Hall, New York, 1993.

[24] A. A. Efros and W. T. Freeman. Image quilting for texture synthesis and transfer. In E. Fiume, editor, *Proceedings of SIGGRAPH 2001*, Computer Graphics Proceedings, Annual Conference Series, pages 341–346. ACM Press / ACM SIGGRAPH, August 2001.

[25] A. A. Efros and T. K. Leung. Texture synthesis by non-parametric sampling. In *IEEE International Conference on Computer Vision*, pages 1033–1038, Corfu, Greece, September 1999.

[26] L. Ferryanto. A kolmogorov-smirnov type statistics for detecting structural changes of textures. *Pattern Recognation Letters*, 16:247–256, 1994.

[27] L. Ferryanto. On estimation of the Walsh-Fourier spectral density of two-dimensional strictly homogenous random fields. *Journal of Nonparametric Statistics*, 5:391–407, 1995.

[28] J. Franke and W. Härdle. On bootstrapping kernel spectral estimates. *Annal of Statistics*, 20:121–145, 1992.

[29] J. Franke, J.-P. Kreiss, and E. Mammen. Bootstrap of kernel smoothing in nonlinear time series. *Bernoulli*, 8,1:1–37, 2002.

[30] V. Friederichs. *Local Smoothing Methods in Image Processing*. PhD thesis, Universitaet Kaiserslautern-Germany, Department of Mathematics, 1999.

[31] T. Gasser, A. Kneip, and W. Köhler. A flexible and fast method for automatic smoothing. *American Statistical Association*, 86,415:643–652, 1991.

[32] D. Geman. Random fields and inverse problems in imaging. In P. Hennequin, editor, *École d'Été de Probabilité de Saint-Flour XVIII, Lecture Notes in Mathematics*, volume 1427, pages 113–193. Springer-Verlag, 1990.

[33] S. Geman and D. Geman. Stochastic relaxation, gibbs distribution, and the bayesian restoration images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6:721–741, 1984.

[34] S. Geman and C. Graffigne. Markov random field image models and their applications to computer vision. In *Proceeding of the International Congress of Mathematicians*, pages 1496–1517, 1987.

[35] J. Geweke. Bayesian treatment of the independent student t linear model. *Journal of Applied Econometrics*, 8:19–40, 1993.

[36] P. E. Gill and M. H. Wright. *Practical Optimization*. Academic Press Inc.LTD, London, 1981.

[37] G. Gimel'farb. Texel-based prototypes of regular mosaics. Website, 2002. `http://www.citr.auckland.ac.nz/~georgy/research/texture/mosaics.html`.

[38] G. Gimel'farb, L. Yu, and D. Zhou. Geometric structure and randomness in texture analysis and synthesis. In *Proceedings of the Seventh International Conference on Digital Image Computing: Techniques and Applications*, pages 10–12, Macquarie University, Sydney, Australia, December 2003. DICTA, CSIRO Publishing.

[39] G. L. Gimmel'farb. *Image Textures and Gibbs Random Fields*. Kluwer Academic Publishers, Dordrecht-Boston-London, 1999.

[40] M. Haindl. Texture synthesis. *CWI Quaterly*, pages 305–331, 1991.

[41] P. Hall and J.D.Hart. Bootstrapp test for difference between means in nonparametric regression. *Journal of The American Statistical Association*, 85,412:1039–1049, 1990.

[42] W. Härdle and A.Bowman. Bootstrapping in nonparametric regresssion: local adaptive smoothing and confidence bands. *Journal of The American Statistical Association*, 83:102–110, 1988.

[43] W. Härdle and E.Mammen. Comparing nonparametric versus parametric regression fits. *The Annals of Statistics*, 21, No.4:1926–1947, 1993.

[44] W. Härdle, E.Mammen, and M. Müller. Testing parametric versus semiparametric modeling in generalized linear models. *Journal of The American Statistical Association*, 93, No.444:1461–1474, 1998.

[45] J. D. Hart. *Nonparametric smoothing and lack-of-fit tests*. Springer, New York, 1997.

[46] D. Heeger and J. Bergen. Pyramid-based texture analysis/synthesis. In *SIGGRAPH 95 Conference Preceedings*, pages 229–238. ACM SIGGRAPH, Addison-Wesley, August 1995.

[47] E. Herrmann, T. Gasser, and A. Kneip. Choice of bandwidth for kernel regression when residuals are correlated. *Biometrika*, 79,4:783–795, 1992.

[48] E. Herrmann, M. Wand, J. Engel, and T. Gasser. A bandwidth selector for bivariate kernel regression. *Journal of Royal Statistics Society series B*, 57,1, 1995.

[49] P. Huber. *Robust Statistics*. Wiley, New York, 1981.

[50] H. Jeffreys. *Theory of Probability*. Clarendon Press, Oxford, third edition, 1961.

[51] J.P.Lesage. Bayesian estimation of spatial autoregressive models. *International Regional Science Review*, 20, 1&2:113–129, 1997.

[52] B. Julesz. Visual pattern discrimination. *IRE Transaction on Information Theory*, IT-8:84–92, 1962.

[53] R. L. Kashyap and R. Chellappa. Estimation and choice of neighbors in spatial interaction models of images. *IEEE Transactions on Information Theory*, IT-29:60–72, Jan. 1983.

[54] E. King, J. Hart, and T. Wehrly. Testing the equality of two regression curves using linear smoothers. *Statistics & Probabiltiy Letters*, 12:239–247, Sept. 1991.

[55] K. Kohrt. Automatische qualitätskontrolle - bildverarbeitung in der industrie. *Keramische Zeitschrift*, 2, April 2005.

[56] S. Lakshmanan and H. Derin. Gaussian markov random fields at multiple resolution. In R. Chellappa and A. Jain, editors, *Markov Random Fields, Theory and Applications*, pages 131–157. Academic Press,Inc., 1993.

[57] S. Lakshmanan, A. K. Jain, and Y. Zhong. Multi-resolution image representation using markov random fields. In *ICIP 94 Conference Preceedings*, pages 855–860. ICIP, 1994.

[58] P. M. Lee. *Bayesian Statistics: An Introduction*. Arnold, London, 2nd edition, 1997.

[59] J. P. LeSage. Spatial econometrics. Website, 1999. `http://www.rri.wvu.edu/WebBook/LeSage/spatial/spatial.html`.

[60] E. Lewis and N. Fieller. A recursive algorithm for null distributions for outliers: I. gamma samples. *Technometrics*, 21:371–376, 1979.

[61] R. Liu. Bootstrap procedures under some non i.i.d models. *The Annals of Statistics*, 16:1696–1708, 1988.

[62] E. Mammen. Bootstrap and wild bootstrap for high dimensional linear models. *The Annals of Statistics*, 21:255–285, 1993.

[63] J. Marron and A. Tsybakov. Visual error criteria for qualitative smoothing. *Journal of The American Statistical Association*, 90:499–507, 1995.

[64] R. Mead. A mathematical model for the estimation of inter-plant competition. *Biometrics*, 23(2):189–205, 1967.

[65] F. Meyer and R. Coifman. Brushlets: A tool for directional image analysis and image compression. *Applied and Computational Harmonic Analysis*, 4:147–187, 1997.

[66] N. Neumeyer and H. Dette. Nonparametric comparison of regression curves: An empirical process approach. *The Annals of Statistics*, 31,3:880–920, 2003.

[67] R. Paget and I. D. Longstaff. Texture synthesis via a noncausal nonparametric multiscale markov random field. *IEEE Transactions on Image Processing*, 7,6:925–931, June 1998.

[68] R. D. Paget. *Nonparametric Markov Random Field Models for Natural Texture Images*. PhD thesis, The University of Queensland, Australia, 1999. Website. `http://www.vision.ee.ethz.ch/~rpaget/publications.htm`.

[69] R. D. Paget and I. D. Longstaff. Texture synthesis and unsupervised recognition with nonparametric multiscale markov random field models. Website, 1997. `http://www.vision.ee.ethz.ch/~rpaget/publications.htm`.

[70] D. N. Politis. Resampling time series with seasonal components. Website, 2001. `http://www.galaxy.gmu.edu/interface/I01/I2001Proceedings/DPolitis/DPolitis.pdf`.

[71] J. Polzehl and V. Spokoiny. Propagation-separation approach for local likelihood estimation. Website, 2004. `http://www.wias-berlin.de/people/polzehl/papers/awslike.2005_1_4.pdf`.

[72] J. Polzehl and V. G. Spokoiny. Adaptive weights smoothing with applications to image restorations. *Journal of Royal Statistics Society Series B*, 62,2:335–354, 2000.

[73] M. Schäl and H. Burkhardt. Automatic detection of errors on textures using invariant grey scale features. In *Workshop on Texture Analysis*, University of Freiburg, Germany, 1998.

[74] M. Scholz. Simulation of textures. Master's thesis, Universitaet Kaiserslautern, Department of Mathematics, 1997.

[75] D. F. G. Schwerpunktprogramm1114. Mathematical methods for time series analysis and digital image processing. Website, 2001. `http://www.math.uni-bremen.de/zetem/DFG-Schwerpunkt/`.

[76] F. S.Cohen, Z. Fan, and S. Attali. Automated inspection of textile fabrics using textural models. *IEEE Transactions on Patten Analysis and Machine Intelligence*, 13,8:803–808, August 1991.

[77] O. Sezer, A. Ertüzün, and A. Erçil. Independent component analysis for texture defect detection. Website, 2000. `http://vpa.sabanciuniv.edu/DD-ICA%20Revised.pdf`.

[78] M. Shimshoni. On fisher's test of significance in harmonic analysis. *Geophys.J.R.Astronom.Society*, 23:373–377, 1971.

[79] S. Siggelkow and M. Schäl. Fast estimation of invariant features. In e. W. Förstner, editor, *Mustererkennung, DAGM*. Springer Verlag, 1999.

[80] G. R. Terrell and D. W. Scott. Variable kernel density estimation. *The Annals of Statistics*, 20,3:1236–1265, 1992.

[81] B. Walsh. Introduction to bayesian analysis, lecture notes. Website, 2002. `http://nitro.biosci.arizona.edu/courses/EEB596/handouts/Bayesian.pdf`.

[82] J. Weickert. *Anisotropic Diffusion in Image Processing*. Teubner, Stuttgart, 1st edition, 1998.

[83] P. Whittle. On stationary processes in the plane. *Biometrika*, 41:434–449, 1954.

[84] P. Whittle. Bounds for the moments of linear and quadratic forms in independent variables. *Theory of Probability and its Applications,*, V(3):302–305, 1960.

[85] L. Y. Wie and M. Levoy. Fast texture synthesis using tree-structured vector quantization. In *SIGGRAPH*, pages 479–488, 2000.

[86] G. Winkler. A stochastic algorithm for maximum likelihood estimation in imaging. Website, 1998. `http://ibb.gsf.de/preprints/1998/pp98-07.html`.

[87] G. Winkler. *Image Analysis, Random Fields and Markov Chain Monte Carlo Methods*. Springer-Verlag, Berlin-Heidelberg, 2003.

[88] C. Wu. Jackknife, bootstrap and other resampling methods in regression analysis. *The Annals of Statistics*, 14,No.4:1261–1295, 1986.

[89] Y. N. Wu, S. C. Zhu, and X. W. Liu. Equivalence of julesz texture ensambles and frame models. *International Journal of Computer Vision*, 38,3:247–265, 2000.

[90] L. Younes. Estimation and annealing for gibbsian fields. *Annales de l'Institut Henri Poincaré*, 24,2:269–294, 1988.

[91] L. Younes. Maximum likelihood estimation for gibbs fields. spatial statistics and imaging. In A. Possolo, editor, *Lecture Notes-Monograph Series, Institute of Mathematical Statistics*. Hayward, California, 1991. Website. `http://www.cmla.ens-cachan.fr/Utilisateurs/younes/publi.html`.

[92] D. Zhou and G. Gimel'farb. Bunch sampling for fast texture synthesis. In N. Petkov and M. A. Westenberg, editors, *Computer Analysis of Images and Patterns, 10th International Conference*, volume 2756 of *Lecture Notes in Computer Science*, pages 25–27, Groningen, The Netherlands, August 2003. CAIP, Springer.

[93] S. C. Zhu, Y. N. Wu, and D. Mumford. Filters, random fields and maximum entropy (frame)-toward unified theory for texture modeling. *International Journal of Computer Vision*, 27,2:107–126, 1998.

Curicullum Vitae

Personal

Name	:	Siana Halim
Place/Date of birth	:	Madiun, 09 November 1970
Nationality	:	Indonesian

Education

1977-1983	-	Elementary school SD Katolik Santa Maria I Madiun, Indonesia
1983-1986	-	Middle high school in SMPN 1 Madiun, Indonesia
1986-1989	-	High school in SMAN 2 Madiun, Indonesia
1989-1993	-	Sarjana Sains (bachelor) in mathematics, Institut Teknologi Sepuluh Nopember Surabaya Indonesia
1996-1998	-	M.Sc in Industrial Mathematics Universitaet Kaiserslautern
Oct.2001- Sept. 2005	-	Ph.D in Department of Mathematics Technische Universitaet Kaiserslautern

Employment

1993 - 1996	:	Assistant lecturer in Industrial Engineering Department, Petra Christian University -Surabaya Indonesia
1998 - 2001	:	Lecturer in Industrial Engineering Department, Petra Christian University -Surabaya Indonesia